Optimization strategies for Markov chain Monte Carlo inversion of seismic tomographic data.

Dissertation

zur Erlangung des akademischen Grades doctor rerum naturalium
(Dr. rer. nat.)

vorgelegt dem Rat der Chemisch-Geowissenschaftlichen Fakultät
der Friedrich-Schiller-Universität Jena

von M.Sc.-Physik, Francesco Fontanini
geboren am 08.11.1983 in Verona (Italien)

Gutachter:

1. **Prof. Dr. Florian Bleibinhaus**
   Lehrstuhl für Angewandte Geophysik
   Montanuniversität Leoben

2. **Prof. Dr. Michael Korn**
   Lehrstuhl für Teoretische Geophysik
   Universität Leipzig

Tag der Verteidigung: 04.07.2016

Contents

Abstract

Probabilistic approach to inverse problem by means of Monte Carlo simulation is a computationally intensive approach whose feasibility has shown to be directly connected with the availability of computational resources and optimization. This study aims to introduce at first some fundamental theoretical aspects and to focus on the issue of optimization of McMC algorithms.
We developed a transdimensional inversion scheme in the framework offered by the established deterministic inversion code `simulr16`. The issues of optimization and performance improvement were tackled by means of parallel independent realizations of the sampling process in addition to a staggered grid approach. The inverse model parametrization of the `simulr16` code in conjunction with transdimensional McMC sampling, provided an affordable and reliable inversion strategy able to offer naturally smooth solutions equipped with a quantitative uncertainty estimation. Our probabilistic inversion method was tested on synthetic data and then applied on the inversion of a field data set from the Salzach valley (Austria). The structures recovered with our approach are compatible with those obtained with other well established methods.
Metropolis-Hastings-based McMC algorithms require a careful tuning in order for the model space to be optimally sampled. Sub-optimal scaling of the size of random walk steps for Markov samplers leads to less efficient chains that require longer runtimes. We proposed a multivariate updating scheme that, using information carried by the model resolution matrix, proved to improve the performances of the classical M-H proposal. Trade-off relations between model parameters were obtained from the model resolution matrix and implemented in our updating scheme. McMC and non-stochastic tests revealed an improvement in performance in terms of increased acceptance rate and enhanced mixing properties.

Zusammenfassung

Der probabilistische Ansatz für die Lösung des Inversionproblems unter Verwendung der Monte Carlo Simulation ist ein sehr rechenintensiver Ansatz dessen Umsetzbarkeit direkt mit den verfügbaren rechentechnischen Ressourcen verbunden ist. Die Absicht dieser Studie ist zuerst einige fundamentale theoretische Aspekte darzustellen und anschlieend das Problem der Optimierung von McMC Algorithmen zu erläutern.
Wir entwickelten ein mehrdimensionales Inversionsschema welches in den bereits vorhandenen deterministischen Inversionscode simulr16 integriert wurde. Das Problem der Optimierung und Verbesserung der Leistungsfähigkeit wurde bewältigt unter Verwendung von parallelen unabhängigen Modelprobenketten und des staggered grid Ansatzes. Die Parametrisierung des Inversionsmodells im simulr16 Code in Verbindung mit der mehrdimensionalen McMC Probenkette ermöglicht eine erschwingliche und zuverlässliche Inversionsstrategie welche natürlich glatte Lösungen mit einer zusätzlichen quantitativen Unsicherheitsbestimmung bereitstellt. Unsere probabilistische Inversionsmethode wurde an synthetischen Daten getestet und anschlieend auf den realen Datensatz des Salzach Tals (österreich) angewendet. Die mit unserer Methode aufgelösten Strukturen sind vergleichbar mit Denen von anderen bereits integrierten Inversionsmethoden.
Metroplolis-Hasting basierende McMC Algorithmen benötigen eine sorgfältige Anpassung, damit der Modelraum optimal abgetastet wird. Eine suboptimale Skalierung der Gröe der zufälligen Laufschritte für Markov-Sampler führt zu geringer effizienten Ketten, welche längere Laufzeiten benötigen. Zur Verbesserung der Leistungsfähigkeit von klassischen M-H-Ansätzen schlagen wir ein multivariates Update-Schema vor, dass die Informationen der Modell-Auflösungsmatrix nutzt. Die Austauschbeziehung zwischen den Modellparametern wird durch die Auflösungsmatrix bereitgestellt und wurde in unserem Update-Schema implementiert. McMC und Non-stochastische Tests zeigen eine Verbesserung in der Leistungsfähigkeit im Sinne von ansteigender Akzeptierungsrate und erhöhter Mischeigenschaft.

List of Figures

List of Tables

List of Abbreviations

PDF ..... Probability Density Function

McMC ..... Markov chain Monte Carlo

rj-McMC ..... reversible jump Markov chain Monte Carlo

ART-PB ..... Approximate Ray Tracing Pseudo ray Bending

ResM ..... Resolution Matrix

RDE ..... Resolution Diagonal Elements

gcd ..... greatest common divisor

iid ..... independent identically distributed

CLT ..... Central Limit Theorem

M-H ..... Metropolis-Hastings

LSQR ..... Least Squares

DLS ..... Damped Least Squares

FATT ..... First Arrival Traveltime tomography

RRTT ..... Refraction-Reflection Traveltime tomography

FWI ..... Full Waveform Inversion

$\mathbf{L_D}$ ..... Likelihood differences

Chapter 1

Bayesian inference and McMC methods: a probabilistic approach to inverse problems in geophysics.

1.1 Deterministic VS probabilistic approach to inverse problems.

Seismic traveltime tomography is, together with other inverse problems in geophysics, often approached through iterative linearized techniques which simplify a non-linear physical reality while aiming to obtain an optimum solution, a single model, found avoiding local minima by means of regularization.
This deterministic approach to the inversion of traveltimes in seismic tomography carries many of the sources of uncertainty and instability that generally characterize inverse problems. Uneven coverage, limited quality of the data, inadequate parametrization, and non-uniqueness of solutions, are within the most notable. Regularization techniques are employed in the attempt to avoid local minima, ad-hoc optimized parameterizations are used but nonetheless the null space remains unknown and the single-model solution generally proposed does not reflect the intrinsic non-uniqueness of the inverse problem. Providing constraints on the uncertainty is therefore a major issue, some of the most widely adopted methods include the evaluation of ray path densities, null-space energy, resolution matrix diagonal elements, and other estimators that provide a qualitative estimation of the uncertainty and on the interdependency of the inverse parameters. A qualitative map of the spatial resolving power of a data set is often obtained with checkerboard tests. A mayor downside of these tests and methods is connected with the intrinsic nature of the deterministic solution they are probing: they are local. No global overview of all the possible solutions and respective uncertain-

ties can be achieved through inverse methods that linearize non-linear physical systems. Despite the mentioned limitations the approach to inverse problems in geophysics is mainly deterministic; still valuable information and interpretation results are proposed and positively utilized.

In contrast to the previous methods, the probabilistic, or Bayesian, approach to seismic tomographic problems is a fully non-linear approach. All the knowledge on the physical objects under study is conveyed in terms of probabilities, the whole model space is analyzed with the positive outcome that local minima aren't disregarded in the quest for a solution, on the contrary the probabilistic approach provides a global overview on the values of the model parameters together with their relative uncertainties. This results in the possibility to quantitatively estimate the non-uniqueness of the problem in terms of probability density functions and to obtain not a single solution but global inference from a stochastic ensemble. Bayesian theory joins a priori information that we have before performing measurements, with the ability of different sets of model parameters to fit the measured data (likelihood), in order to obtain a conditional probability density function (PDF) in the model space, referred as *posterior* distribution.

The probabilistic approach to inverse problems offers a further advantage: the possibility to treat the number of model parameters as an unknown in the inversion process allowing the data to drive the parametrization. This achievement seems to remove that source of uncertainty given by the necessary choice or estimation of many inversion parameters. The task to define number and distribution of parameters as well as damping and smoothing is in this way passed to the data itself, reducing the possible error sources due to this potentially subjective choice. Nonetheless, depending on the way specific algorithms are implemented, other different parameters might be introduced in the inversion process, which are also in need of a correct estimation. See for instance the treatise of Bodin et al. (2012) of unknown data noise as an hyperparameter in rj-McMC. Adaptive and irregular parametrization strategies are however established also in the deterministic framework (see section 1.3.2), together with methods and workflows to assess optimal values for some inversion parameters.

1.2 Probabilistic methods in geophysics: state of the research

The development of probabilistic algorithms is strongly linked to the development of fast computing machines. It was pioneered by Metropolis et al. (1953), who developed and applied a Markov Chain algorithm to investigate the Boltzmann distribution. His approach was generalized a few years later by Hastings (1970).

First geophysical applications of probabilistic methods to inverse problems were reported by Press (1968), Keilis-Borok and Yanovskaja (1967), and during the following decades, Monte Carlo methods became an established inversion approach for small geophysical problems. Geophysical applications of Bayesian inference are described in Tarantola and Valette (1982), Duijndam (1988a,b), Mosegaard and Tarantola (1995). A brief overview of the early applications and the development of probabilistic techniques are given in the review paper of Sambridge and Mosegaard (2002). Today, probabilistic methods are well established for the 1D inversion of body wave traveltimes (Sambridge and Mosegaard, 2002), and are also widely used for the 1D inversion of surface wave dispersion curves at a broad range of scales (Shapiro and Ritzwoller, 2002; Socco and Boiero, 2008). As Mosegaard and Tarantola (1995) point out, Markov chains alone may not suffice to cope with realistic geophysical problems, because the acceptance rate of randomly perturbed models can be so low that the problem becomes computationally intractable. The strategy is to restrict the application of the forward solution so far as possible to the relevant models with the practice of importance sampling (Mosegaard, 1998). A common approach is the Markov chain Monte Carlo (McMC) method, mostly implemented through the Metropolis-Hastings algorithm. The problem of the model parametrization has been addressed in a transdimensional framework mostly with the reversible-jump algorithm by Green (1995); Green and Mira (2001) (Sambridge et al., 2006; Gallagher et al., 2009). Most of the published works employing rj-McMC algorithms lie in the genetics field. Bodin et al. (2012) implemented a self-parametrized data noise treatment as an extension of their delayed-rejection, reversible-jump algorithm (Bodin and Sambridge, 2009).

1.3 Computational cost and optimization strategies.

Markov chain and McMC methods are extremely computationally-intensive algorithms which generally require computation times that reach some order of magnitude more than their deterministic equivalent. For this reason a number of strategies must be considered in order to contain the computational time. In the following sections a brief overview is presented on the most commonly adopted strategies.

1.3.1 Forward modeling

The solution of the forward problem is often the part of the inversion process that absorbs most of the CPU time. Seismic tomography problem are not an exception and the optimization of the forward routines is fundamental to avoid

wasting computational resources. In the framework of the software package we are developing, a choice is given to the user on the forward computation strategy to adopt: either a bending ray tracer combined with a grid search of Bleibinhaus (2003) after Um and Thurber (1987)or a finite-differencies-eikonal solver of Vidale (1990) with modifications of Hole (1992). In this work the latter will be always employed to solve the forward problem. The CPU time spent in the computation of the time-field is directly proportional to the number of seismic sources; a test performed on a synthetic dataset (see section 2.3) characterized by 23 sources shows that forward computation through the eikonal solver takes over 90% of the time needed to perform an iteration. Parallelization of the forward routines over the sources is therefore a valid approach that promises a reduction of the computing time almost proportional to the number of CPUs/cores utilized. Additional strategies intended to reduce the forward time-cost have been developed and they include emulations and approximations of the forward modeling. Dębski (2010) for instance simply uses straight ray paths for a 2D inversion of body waves while Bodin and Sambridge (2009) use great-circle paths for the 2D inversion of surface wave group velocity for Australia. Such simplifications can be justified, if the actual rays are close to those paths, and if moderate velocity perturbations cause only minor ray path deviations. In general, approximate forward modeling is acceptable in a probabilistic framework (Koutsourelakis, 2009).

1.3.2 Model Parametrization

The computational load of probabilistic methods can be reduced limiting the number of the inverse variables that parametrize a certain model. A number of studies deal with methods to adapt the inverse grid, employing irregular parametrization schemes aiming to match the resolving power of the data. Different strategies have been proposed and applied to seismic inverse problems by, e.g., Sambridge et al. (1995), Thurber and Eberhart-Phillips (1999), Böhm et al. (2000), Bleibinhaus (2003), Trinks et al. (2005), Ajo-Franklin et al. (2006), Bleibinhaus and Gebrande (2006) and Bodin and Sambridge (2009). In our work in Chap.2 we will adopt a transdimensional approach allowing the number of inverse parameter to vary during the sampling process thus becoming part of the set of variables. This approach results in a parameterization that is determined by the data itself, with the counter-intuitive outcome that over-parametrized models are naturally discouraged without any preference for simpler models being expressed (Sambridge et al., 2006). Such a property of transdimensional Bayesian inference is referred to as principle of *natural parsimony*. In the optic of optimization of McMC inversion schemes we combined the transdimensional approach with the use of staggered grids: a strategy that combines the advantages of limited-resolutions grids with the desired high resolution of seismic data processing (Böhm et al., 2000).

1.3.3 Optimized updating schemes

Markov chain Monte Carlo sampling through the Metropolis-Hastings algorithm demands a properly tuned choice of proposal distribution in order to achieve good efficiency. Automatic tuning and scaling of proposals can be obtained through Adaptive McMC, yet this approach requires specific attentions to preserve important properties of the chain (see eq.3.22). Often more trial and error and heuristic approaches are employed to reach *ad hoc* optimal scaling. In this study the proposals will be scaled according to Gelman et al. (1996) aiming to maintain the acceptance rates between 20 and 30%. Nonetheless Rosenthal points out that the algorithms efficiency remains high whenever the acceptance rate lays in the range $10 - 60\%$ (Brooks et al., 2011).
A further strategy that can be applied in a McMC approach considers the use of multivariate updating schemes that propose to update more than just a single inverse parameter at a time. If on one side multivariate schemes have the potential to reduce the computational load increasing the step length of the random walk in the model space, on the other side they often result in a lowered acceptance ratio with the consequence that no improvement is observed in the mixing properties of a Markov chain implemented with such a scheme. A strategy to face the problem of increased probability of rejection will be presented in Chapter 3 where we propose a multivariate updating scheme that attempts to propose "better" models exploiting the information carried by the model Resolution matrix.

1.3.4 Parallelization of Markov processes

Markov chains are intrinsically serial stochastic processes and despite contradictory opinions in the literature many approaches to parallelization have been proposed and applied. In a debate on the use "one long run *VS* many short runs" it has been pointed out (Geyer, 1991, 1992) that a number of pitfalls could be present in parallel approaches, first of all the lack of convergence or the pseudo-convergence of *short-run-chains*. Short runs could also lead to an increased difficulty in the detection of coding bugs. In spite of all the contrary argumentations many are the examples in literature where inference was made using several independent sequences (Gelman and Rubin, 1992) and many are the "embarrassingly parallelisable" McMC algorithms (Rosenthal, 2000). The first, more direct approach is parallel computing through multiple independent Markov chains. A number of McMC instances are launched each with a different initial state, after all the chains reached convergence the ensembles are subsampled and joined in a single set that retains the properties of the single chains and has the same equilibrium distribution. In Chapter 2 we opted for this approach in our transdimensional code where we chose to initiate and then join independent Markov chains. Examples of other

more complex approaches to parallel computing include Metropolis-coupled McMC (Geyer, 1991), Simulated Tempering (Marinari and Parisi, 1992; Geyer, 1991) and Population McMC (Laskey, 2003). Some recent algorithms to parallelize independent or interactive chains chains are proposed respectively by VanDerwerken and Schmidler (2013) and Campillo et al. (2009). Recent geophysical applications on parallel McMC computing are a Parallel Tempering algorithm for probabilistic sampling and multimodal optimization (Sambridge, 2014).

1.4 McMC within Simulr16

For the studies reported in this thesis work we developed our own McMC algorithms as modules of an established deterministic-inversion software, allowing the use of pre-existing routines and a seamless integration with the original deterministic inversion scheme.

A Bayesian-inversion algorithm has been implemented and integrated in the simulr16 code by Bleibinhaus (2003) that can invert refracted and reflected travel time data for velocities, hypocenters, station delays and reflector positions simultaneously. It is based on simulps12 and simulps13q originally created by Thurber (1983) and developed by Um and Thurber (1987), Eberhart-Phillips (1986) and Rietbrock (1996). The forward computation of travel times can be performed choosing between an approximate-ray-tracing pseudo-ray-bending (ART-PB) algorithm from Thurber C. H. (1987) or an eikonal solver from Hole and Zelt (1995). For the applications reported in this paper travel times have been computed only with the eikonal solver. The parametrization is based on a 2D/3D grid of velocity nodes defined by the intersections of orthogonal planes with irregular plane-spacing. This generally irregular inverse grid is based on node properties which define the parameters as inverted, interpolated, linked or fixed, thus obtaining a regular rectilinear grid to map velocities.

An introductory schematic description of McMC algorithms can be given through a four-phases workflow:

1. **Random walk in the model space**: trial models are sampled generating their parameters drawing from a prior probability distribution. At every step of the inversion a new model is proposed randomly perturbing some parameters of the previous one. The kind of perturbation to be applied is selected with a defined probability, then a node to be perturbed is randomly chosen:

    - *Velocity perturbation*: a new value of velocity is chosen from a gaussian

probability density centred on the previous value.

- *Trans-dimensional perturbation*: an existing parameter can be removed from the set of inverse parameters or vice versa a new one can be generated and added. This step is often referred to as *birth-death step.*

2. **Forward modelling**: travel times computation and evaluation of the likelihood for the proposed model

3. **Metropolis-Hastings step**: proposed trial models are accepted or rejected with a probability that depends on their ability to reproduce observations.

4. **Ensemble analysis**: estimation of the statistical properties of the Markov chain formed by the collected models. This is not performed during the runtime by the main code. A separate program can perform all the statistical analysis and computations on the ensemble at any stage of the sampling process.

At this point it is important to point out that the above described algorithm represents a general workflow that could apply to both transdimensional algorithms as in Chapter 2 and non-transdimensional ones, as the ResM-based McMC that will be presented in Chapter 3. For non-transdimensional algorithms the probability mentioned in step 1 will be set to 0, in this way the models in resulting chain will undergo velocity perturbations only. The way this kind of perturbation is performed (transition kernel) can be defined differently: one could perturb one single model parameter, or multiple ones at a time (multivariate perturbation scheme). Specific algorithm schemes will be described and discussed in detail; while the transition kernels of the algorithms presented in Chapters 2 and 3 will have substantial differences, the general four-stages workflow illustrated above is going to be preserved.

1.5 Statistical inference: from integration to Markov chain Monte Carlo

"*Markov chain Monte Carlo (McMC) is a technique for estimating by simulation the expectation of a statistic in a complex model. Successive random selections form a Markov chain, the stationary distribution of which is the target distribution. It is particularly useful for the evaluation of posterior distributions in complex Bayesian models. In the Metropolis-Hastings algorithm, items are selected from an arbitrary proposal distribution and are retained or not according to an acceptance rule.*"

This abstract of the *Encyclopedia of Biostatistics* from Gilks (2005) is a short but pregnant excursus that yields some of the most basic, yet fundamental, concepts of Bayesian inference. We will try in this introductory section to give a quick overview of the path that leads from statistical inference to the McMC methods that will be used in this thesis.
The two major classes of numerical problems in statistical inference are *optimization* and *integration* problems, to the latter we can generally associate Bayesian inference (Robert and Casella, 2004, pp. 71) in the form of McMC methods, while optimization problems will not be part of this thesis. The use of Monte Carlo simulation to solve numerical integration problems comes in handy when the "curse of dimensionality" leads deterministic numerical integration methods to fail or to lack of efficiency. The number of function evaluations needed for an adequate accuracy grows in fact exponentially with the number of variables, making high-dimensional functions virtually unmanageable for deterministic numerical integration. Monte Carlo methods provide an alternative to this issue trying to evaluate the integral of the function of interest $h(x)$ by means of a *density* function $\pi(x)$:

$$E_\pi\left[h(X)\right] = \int h(x)\pi(x)dx \tag{1.1}$$

The use of $\pi(x)$ to increase the density of the samples where the integrand is larger, known as *importance sampling*, aims to optimize the evaluation process. In order to know an appropriate density function one should already know the integral, or alternatively approximate it with a function of similar distribution or with adaptive routines. When sampling from relatively simple distributions, Monte Carlo algorithms take a large sample of random variables (which can of course be vectors) and then compute the average of h on that sample.
Markov chain Monte Carlo methods come into play to solve the problem of sampling from complicated unknown distributions where the average of h computed on the sampled variables doesn't approximate h well enough. If instead of gener-

ating statistically independent samples $(X_1, \ldots, X_n)$ we generate them correlated, with specific probabilities for the system to move between states, our random walk assumes the characteristics of a *random walk on a graph*, which in fact is a Markov chain. Some special kind of Markov chains have the fundamental probability of having a *stationary distribution*, concept that can be simplistically explained saying that the probability for a very long random walk to end up to some particular state is independent from the starting point of the random walk. Such probability is also unique. Some brief formal support for this fundamental theorem of Markov chains will be provided in the next section where the condition for the existence and unicity of a stationary distribution will be outlined.
Going back to the problem of integration, where we aim to produce samples characterized by the density function $\pi(x)$, the McMC strategy is to generate a Markov chain with exactly the desired distribution and prove that the convergence is relatively quick in comparison to the dimension of the state space. This special kind of random walk on a graph is accomplished with methods as Gibbs sampler (Martin A. Tanner, 1987), Sequential Monte Carlo, also known as Particle Filter (Del Moral, 1996), and Metropolis-Hastings, which is the one that will be utilized in this work.

1.6 Fundamental properties of Markov Chains

In this section we aim to collect some basic formal definitions and theorems that are part of the theoretical basis of the Markov chain theory. The practical reason for this formal section is to lay down formal justifications for some properties of McMC (scheme in Fig.1.1), fundamental for the treatment that will be given in the next chapters. The concepts we are going to synthetically describe can be found in literature, where a number of authors give extensive and exhaustive attention to stochastic processes and Markov chains. For a complete and more formal treatise a suggest source is "Monte Carlo Statistical Methods" (Robert and Casella (2004), whose formalism is used in this chapter), while the "Handbook of Markov Chain Monte Carlo" (Brooks et al., 2011) focuses on a more hands-on approach and will be often used as a guideline and source in this thesis.

1.6.1 Markov Chain

Definition 1.1 (**Stochastic process**) A collection of aleatory variables $(X_t)_{t\in T}$ is called *stochastic process*. If in particular $t \in \mathbb{N}$, then such a collection takes the name of *discrete-time stochastic process* and it's written as $(X_t) = (X_0, X_1, X_2 \ldots)$.

The sequential evolution of stochastic processes is usually described in terms of time: considering a variable X_N, the *actual state* of a stochastic process, the subset

$(X_0, \dots, X_{N-1})$ is called *past*, while similarly the *future* states are those belonging to the subset $(X_{N+1}, \dots)$.

Definition 1.2 (**Markov chain**) A Markov chain is a particular kind of stochastic process with values in a state space S. For what concerns this study we will assume that:

- The set T is always countable, therefore we will always consider discrete-time stochastic processes; we can assume that $T = \mathbb{N}$ since for what concerns us T should simply represent the successive iterations, thus the temporal evolution of our McMC inversion algorithm.
- The set S is generally a subset of $\mathbb{R}^+$, since it represents the support of the parameters vector, we can assume it being discrete for ease of notation.

A Markov chain is a stochastic process in which, known the actual state, past and future are independent, the probability of the chain to move from a state n to a state $n+1$ is conditional only on the actual state. Formally this characteristic can be expressed through the *Markov property*

$$P(X_{n+1} \in A | x_1, x_2, \dots, x_n) = P(X_{n+1} \in A | x_n), \quad A \in S \tag{1.2}$$

In general the above property depends on x, A and n. When there is no dependence on n then the chain is said to be *time-homogeneous*. In such a case we can define a function, called *transition kernel*, $P(x, A)$ based on the following properties:

- $P(x, A)$ defines a probability density on the state space S for all $x \in S$;
- the function $x \mapsto P(x, A)$ is measurable, it can be evaluated for all $A \in S$

In the case where the state space is discrete, $S = \{x_1, x_2, \dots\}$ then P is a transition matrix where the element $p_{i,j}$ is given by $P(x_i, x_j)$; such a matrix is stochastic, thus each row sums up to 1. If S is finite and has r elements, then the transition matrix can be written as:

$$P = \begin{pmatrix} P(x_1, x_1) & \cdots & P(x_1, x_r) \\ \vdots & \ddots & \vdots \\ P(x_r, x_1) & \cdots & P(x_r, x_r) \end{pmatrix} \tag{1.3}$$

The transition matrix defines the transition probabilities between all the possible states during the evolution of a Markov chain, which are defined by conditional probabilities. Using the Chapman-Kolmogorov equations (Robert and Casella, 2004, pp.144) we can express the probability of moving from an initial state X_0 to another state in n moves as:

$$P^n(X_0, A) \equiv P(X_n \in A | X_0) \tag{1.4}$$

which will be useful to introduce the concept of *equilibrium* (or *limit*) distribution: the probability distribution to which a Markov chain converges in limit after a number n of moves (see.eq 1.7). Let us now briefly provide some mathematical support in order to clarify under which conditions π is a stationary distribution and why this is so important for McMC algorithms.

1.6.2 Ergodicity and Stationarity

As introduced above, a fundamental aspect of Markov chains applied on simulation is the study of the asymptotic behavior of the chain while the number of iterations tends to infinity.

Definition 1.3 (**Stationary distribution**) Given a Markov chain (X_n) with a state space S and transition probability $P(x, y)$ then a distribution π is called a *stationary distribution* for (X_n) if it satisfies the condition:

$$\sum_{x \in S} \pi(x) P(x, y) = \pi(y) \qquad \forall y \in S \tag{1.5}$$

Let us now introduce some definitions, useful in the classification of the states of a Markov chain, necessary to determine the nature of the chain itself.

Definition 1.4 (**Irreducibility**) A Markov chain is called *irreducible* if for any two states x, y we have an integer n such that $P^n(x, y) > 0$. This means that it's always possible to move between two states of the chain with transitions of positive probability.

Definition 1.5 (**Periodicity**) Let $x_i \in S$ be a state of a Markov chain with transition matrix P. Indicating the greatest common divisor of some numbers $a_1, a_2, \dots$ with $gcd\{a_1, a_2, \dots\}$, we can define the *period* $d(x_i)$ of the state x_i as:

$$d(x_i) = gcd\{n \geq 1 | P^n(x_i, x_i) > 0\} \tag{1.6}$$

A state for which $d(x_i) = 1$ is called *aperiodic*, a Markov chain where all the states have period 1 is called *aperiodic*.

Definition 1.6 (**Ergodicity**) A Markov chain that holds the property of being both irreducible and aperiodic is said to be *ergodic*.

Using the definition of a stationary distribution given in eq.(1.5) we can say that if a Markov chain has *limit distribution*, that is a distribution π such that

$$\lim_{n\to\infty} P^n(x,y) = \pi(y) \tag{1.7}$$

then π must be a stationary distribution.

Let us now define the notions of recurrence and variational distance between distributions that will come in handy for the enunciation of the theorem that sums up our dissertation on stationarity.

Definition 1.7 (**Recurrence**) A Markov chain defined by a transition kernel P with stationary distribution π is *recurrent* if the average number of visits to an arbitrary set A is infinite, independently from the starting state X_0:

$$P(X_1, X_2, \ldots, \in A|X_0) > 0 \qquad \forall X_0 \tag{1.8}$$

furthermore in case the probability (1.8) for the chain to return an infinite number of time to states in A is $= 1$, then it is *Harris recurrent*.

Definition 1.8 (**Variational distance**) Given $\lambda = (\lambda_1, \ldots, \lambda_k)$ and $\nu = (\nu_1, \ldots, \nu_k)$ measures of probability on the state space S, we can define the *total variation distance* between them as

$$\begin{aligned} \delta(\lambda,\nu) &= \sup_{A\subset S} |\lambda(A) - \nu(A)| \\ &= \frac{1}{2}\sum_{i=1}^{k} ||\lambda_i - \nu_i|| \end{aligned} \tag{1.9}$$

Theorem 1 If a Markov chain with state space S is irreducible and recurrent, then its stationary distribution π is unique, if furthermore the chain is ergodic then it admits a limit distribution corresponding with π

$$\lim_{n\to\infty} P^n(x,y) = \pi(y) \qquad \forall x, y \in S \tag{1.10}$$

that, utilizing the definition of total variation distance, can be rewritten as:

$$\lim_{n\to\infty} \|P^n(x,y) - \pi(y)\| = 0 \qquad \forall x, y \in S \tag{1.11}$$

In other words an ergodic Markov chain has a stationary distribution, and such a distribution is unique.

This theorem can hold its validity also under less restrictive conditions, more specifically the existence of a stationary distribution can be proved also after removing both the irreducibility and aperiodicity conditions, while the unicity of the stationary distribution is maintained only if the irreducibility condition holds (Levin et al., 2006).

Definition 1.9 (**Average**) The average of a probability function $h(x)$ defined on the space state $S = \{x_n\}$ of a Markov chain is:

$$\hat{h}_n = \frac{1}{n} \sum_{i=1}^{n} h(x_n) \tag{1.12}$$

With the concept of average for a probability well defined, it is possible now to enunciate two fundamental theorems:

Theorem 2 (**Ergodic theorem**) Given an ergodic Markov chain with states x_n and stationary distribution π, if h is a function of finite variance such that $E_\pi [h(x)] < \infty$, then:

$$\lim_{n\to\infty} \hat{h}_n = \int h(x)\pi(x)dx = E_\pi [h(x)] \tag{1.13}$$

It is possible to observe that the ergodic theorem is an equivalent for Markov chains of what the strong law of large numbers is for i.i.d. samples since it states that the sample average of the states of a Markov chain is a consistent estimator of the expected value of the limit distribution π, even if the states are statistically dependent. In other words the Ergodic theorem (or Convergence theorem) states that if an irreducible and aperiodic Markov chain is left evolving for a sufficiently long time, then independently from the initial distribution, the marginal distribution of the chain at time n will converge in total variation to the stationary distribution π. For what concerns the application we will made of the Markov chains theory, the importance of the ergodicity of a chain is strongly connected with the possibility to use the sampled models to compute expectations of some function of choice (i.e. mean, mode, errors).
Since the states in a Markov chain generally show a statistical dependence we need the central limit theorem (CLT) to be formulated in order to be able to monitor the convergence expressed by the ergodic theorem.

Theorem 3 (**Central Limit theorem**) If $X = \{X_0, X_1, \dots\}$ is a uniformly (geometrically) ergodic Markov chain then:

$$\lim_{n\to\infty} \sqrt{n} \left(\hat{h}_n - E_\pi [h(x)] \right) = N(0, \sigma_\pi^2) \tag{1.14}$$

where $\sigma_\pi^2 = var_\pi \{h(X_0)\} + 2 \sum_{i=1}^{\infty} cov_\pi \{h(X_0), h(X_i) < \infty\}$

In simpler words the CLT states that, under the condition of uniform ergodicity, the sample average of a sufficiently large set of states will eventually converge to a normal distribution, given a well defined expected value and asymptotic variance.

1.6.3 Reversibility

Considering a discrete-time homogeneous Markov chain $X = \{X_1, \ldots, X_n\}$ with transition matrix $P(x, y)$ and stationary distribution π we might want to study the succession of its states in reverse order: it can be proved that also the succession $X = \{X_n, \ldots, X_1\}$ defines a Markov chain.

Definition 1.10 (**Detailed balance**) A Markov chain is called *reversible* if the distribution of X_{n+1} conditionally on $X_{n+2} = x$ is the same as the distribution of X_{n+1} conditionally on $X_n = x$, such a chain satisfies the *detailed balance* condition:

$$\pi(x)P(x, y) = \pi(y)P(y, x) \qquad \forall x, y \in S \tag{1.15}$$

The importance of reversible Markov chains can be easily explained: if there is a distribution π that satisfies eq.(1.15) for an irreducible Markov chain, then the chain is also positive recurrent, which means that π is also a stationary distribution. Verifying the condition of aperiodicity leads then to the conclusion that π is also a limit distribution. In order to generate a chain with given limit distribution π it is therefore needed to find suitable transition probabilities $P(x, y)$ that follow eq.(1.15). This is accomplished, as already stated, by means of the Metropolis-Hastings algorithm. A graphical summary of the most important properties of M-H-based algorithms is reported in Fig.1.1.

1.7 Markov chain Monte Carlo

Markov chain Monte Carlo (McMC) methods are a family of sampling algorithms particularly useful to deal with target distributions that cannot be directly sampled from. Assuming that one needs to create samples from a target distribution π which can be evaluated but not simply sampled, then a solution is to construct a Markov chain that has π as a limit distribution, and with a sufficient number of steps it will converge to the target distribution. The main application of McMC methods is to make possible, or ease inference in a Bayesian context, where the target distribution π is the posterior distribution of a set of parameters of interest. In geophysical applications, as seismic tomography, this set corresponds to the model parameters we aim to describe statistically.

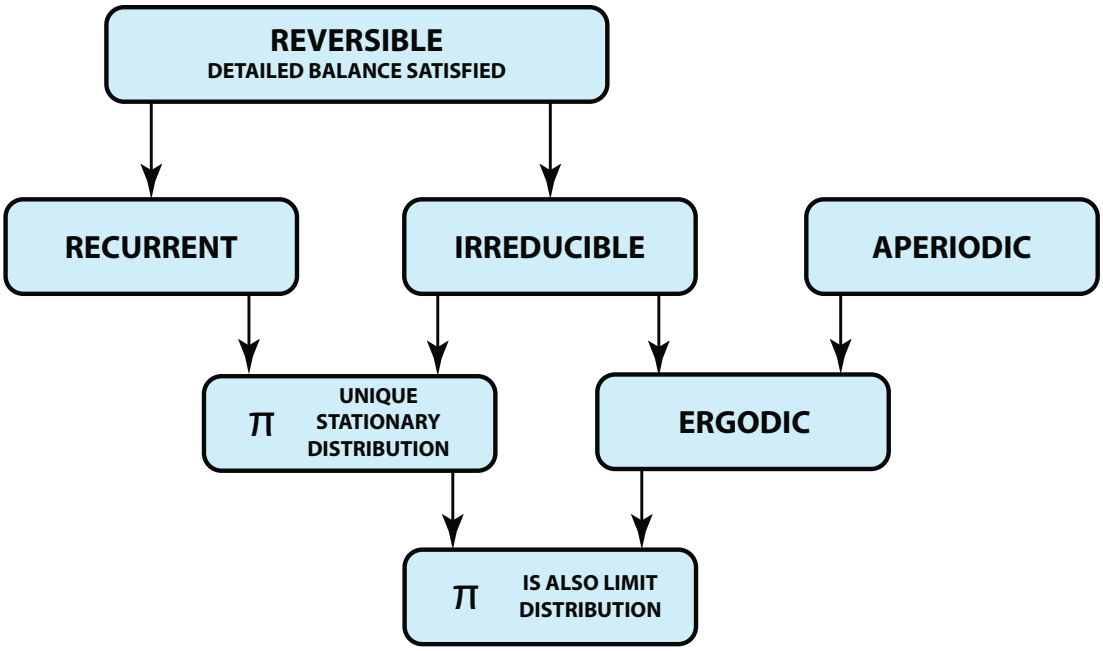


Figure 1.1: *Summary of the properties, and their interrelationships, of the Markov chains involved in this study. Such properties are granted by transition kernels provided by the Metropolis-Hastings algorithm.*

1.7.1 Bayesian inference

Bayesian inference is an approach to statistical inference in which probabilities are not interpreted as frequencies, proportions or any other deterministic concept, but are rather considered as confidence levels for the occurrence of a certain event. Bayes' theorem is the nucleus of this probabilistic method of inference, but before enunciating it let us lay out some basic notation for inverse and forward problems as well as some basic probability concepts for bayesian inference.

1.7.1.1 Inverse problem

Considering a continuous physical system, one can discretize and describe it through a set of model parameters $\mathbf{m} = \{m_1, m_2, \dots\}$, using the available knowledge (expressed with physical laws and theories) it is then possible to compute the data $\mathbf{d}^{cal} = \{d_1^{cal}, d_2^{cal}, \dots\}$ that is expected to be observed from measurements on the given system. This process is defined as *forward problem* and can be expressed as:

$$\mathbf{d}^{cal} = \mathbf{G}(\mathbf{m}) \tag{1.16}$$

Where G is the *forward operator* connecting data and model parameters through a physical theory. The opposite process where one tries to obtain the values of the model parameters given some observed data $\mathbf{d}^{obs} = \{d_1^{cal}, d_2^{cal}, \dots\}$ obtained through actual measurements is the *inverse problem*:

$$\mathbf{m} = \mathbf{G}^{-1}(\mathbf{d}^{obs}) \tag{1.17}$$

The majority of non trivial geophysical inverse problems have non-linear properties: the systems that belong to this category are so complex that generally can be only numerically solved. In order to obtain an analytical solution two possible strategies are available: one could simplify the physical theory involved by means of linear approximations, or could systematically look for a number of possible solutions that fit the data to an acceptable level. The latter strategy is fully non-linear and methods that belong to this family are generally referred to as *Probabilistic*.
Considering now the linearized approach, inverse problems can be categorized by means of the Rouch-Capelli theorem, in terms of how the model parameters are defined through the data. Relating N, the number of model parameters, to the rank of the augmented matrix $(\mathbf{G}|\mathbf{d})$, linear inverse problems can be grouped as follows:

- *Over-determined* problems: $rk(\mathbf{G}|\mathbf{d}) > N$, it is inconsistent, there's no solution if not through approximation techniques as the classical Least Squares regression (LSQR)
- *Determined* problems: $rk(\mathbf{G}|\mathbf{d}) = N$, there is exactly one single solution;
- *Under-determined* problems: $rk(\mathbf{G}|\mathbf{d}) < N$, there is a potentially infinite number of solutions;
- *Mixed-Determined* problems: it is not possible to make any statement about $rk(\mathbf{G}|\mathbf{d})$, some model parameters might be over- others under-determined.

Seismic tomography usually deals with the last group: mixed-determined problems where $\mathbf{G}^{-1}$, the inverse of the forward operator, cannot be computed. In this case regularization techniques as damped least squares (DLS) are employed to obtain a linearized solution. For DLS inversion methods $\mathbf{G}^{-1}$ is substituted with a generalized inverse operator:

$$\mathbf{G}^{-g} = \left[\mathbf{G}^T\mathbf{G} + \theta^2\mathbf{I}\right]^{-1}\mathbf{G}^T \tag{1.18}$$

Where θ is the damping factor, a trade-off parameter that weights the relative importance of errors and solution norm (Gubbins, 2004). The single solution obtained

in this fashion does not reflect the uncertainties and intrinsic non-uniqueness of the problem itself and doesn't account for possible multimodality of the parameters' distribution. Bayesian inference comes into play as a possible answer to the non-uniqueness issue: if an analytical formulation of the solution is not available, then it is possible to express it by means of a probability distribution, that can be estimated through Markov chain Monte Carlo sampling.

1.7.1.2 Probability

Probability is the fundament of statistics and therefore of Bayesian theory; while a complete treatise, formalizing the difference between probability and probability densities, can be found in literature (e.g. Tarantola, 2005 or Menke, 2012), it is however appropriate to recall that probabilities and probability densities can be either:

- *Marginal*: the probability of a single even to occur, without any conditional relation with other events. It can be considered as an unconditional probability. The usual expression for the probability of an event A to happen is $p(A)$.
- *Joint*: the probability of two or more events to happen simultaneously. It is the intersection of the probability for a number of events often written as $p(A \cap B)$, in Bayesian theory however the usual notation is $p(A, B)$, notation that will be in use also in this thesis. The relation between marginal and joint probabilities for two events is: $p(A, B) = p(A)p(B)$
- *Conditional*: the probability of a certain event to occur, given the occurrence of another event. The conditional probability of A, given B is usually written as $p(A|B)$.

In the case studies we are dealing with in this work, the random variables (i.e. p-waves velocity) are sampled on continuous subsets of $\mathbb{R}^+$. In order to provide a graphical representation of probability distributions, such subsets are discretized (in bins of 0.05 Km/s width) while analyzing the sampled ensemble, allowing the definition of a probability, which is normalized to 1 for every depth interval considered in the representation of probability density functions.

1.7.1.3 Bayes' theorem

Bayesian approach addresses the problem expressing all information in terms of probability distributions, the knowledge available on the system before measuring the data is referred to as *a priori* or *prior* probability density. Using a vectorial

notation, the probability of observing the data $\mathbf{d}_{jobs}$ given a model $\mathbf{m}$ is expressed with a *likelihood* function $p(\mathbf{d}_{obs}|\mathbf{m})$ that measures the level of fit between measurements and predictions made using the model $\mathbf{m}$. Prior and likelihood are combined in *Bayes' theorem*:

THEOREM 4

$$p(\mathbf{m}|\mathbf{d}_{obs}) = \frac{p(\mathbf{d}_{obs}|\mathbf{m})p(\mathbf{m})}{p(\mathbf{d}_{obs})} \tag{1.19}$$

where the conditional $p(\mathbf{m}|\mathbf{d}_{obs})$ is the *a posteriori* (or *posterior*) probability density function, which we can refer as the solution of the inverse problem in the Bayesian framework. The denominator term in eq.(1.19) is the *evidence*, a normalizing factor for the posterior in the form $p(\mathbf{d}) = \int p(\mathbf{d}|\mathbf{m})p(\mathbf{m})d(\mathbf{m})$. Since the evidence is not depending on any particular model $\mathbf{m}$ it is often regarded as a constant simplifying thus Bayes theorem in the form:

$$p(\mathbf{m}|\mathbf{d}_{obs}) \propto p(\mathbf{d}_{obs}|\mathbf{m})p(\mathbf{m}) \tag{1.20}$$

or in a simpler explicit notation:

$$posterior \propto likelihood \times prior \tag{1.21}$$

Bayes' theorem guarantees that sampling the model space with the joint informations given by prior and likelihood we can generate samples from a distribution that approximates the posterior distribution. This can be achieved with McMC sampling, provided that the Markov chains are implemented respecting the conditions of being aperiodic and irreducible. What we practically seek is an *update mechanism* (i.e. an algorithm that generates pseudorandom perturbations to a state of a chain) that preserves the stationary distribution we are interested in. Two are the most used algorithms used in the framework of McMC simulation:

- Gibbs sampler
- Metropolis-Hastings algorithm

In this study we will be making exclusive use of the latter, thus no further mention will be given of Gibbs sampler.

1.7.2 Metropolis-Hastings algorithm

Metropolis-Hastings algorithm finds its origin in the original paper of Metropolis et al. (1953) which applied the algorithm on the canonical ensemble, creating samples of the Boltzmann distribution. The idea was afterwards developed by Hastings (1970) which incorporated it in the framework of Markov chain sampling.

M-H algorithms are constructed on appropriate transition kernels $p(x, y)$, following the *detailed-balance condition*$\pi(x)q(x, y) = \pi(y)q(y, x)$, which grant reversibility, sufficient condition to ensure that π is a stationary distribution. The kernel is chosen such that:

$$p(x, y) = q(x, y)\alpha(x, y) \qquad if x \neq y \tag{1.22}$$

where $q(x, y)$ is an arbitrary transition kernel between the current state x and a proposed state y and $\alpha(x, y)$ is defined as an *acceptance probability*. Since there is a positive probability for the chain to remain in x we have

$$p(x, x) = 1 - \int q(x, y)\alpha(x, y)dy \tag{1.23}$$

and consequently

$$p(x, A) = \int_A q(x, y)\alpha(x, y)dy + \mathbb{1}_A \left[1 - \int q(x, y)\alpha(x, y)dy\right] \tag{1.24}$$

for every subset A of the model parameters space. The acceptance probability $\alpha(\cdot, \cdot)$ is chosen such that the resulting chain is reversible, thus:

$$\alpha(x, y) = \min\left\{1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}\right\} \tag{1.25}$$

Before carrying on, let us summarize with a few remarks:

- all McMC algorithms based on Markov chains with transition kernels (1.24) and acceptance probabilities (1.25) are called Metropolis-Hastings McMC;
- the choice of the transition kernel $q(\cdot, \cdot)$ is arbitrary, and it provides a flexible tool in the construction or modification of an algorithm;
- the demonstration that the detailed balance condition is satisfied with the choice of p in eq.(1.24), and therefore defines a reversible chain with equilibrium probability π, follows directly from the definition of acceptance probability given in eq.(1.25).

Using Bayes' theorem (eq.1.19) we can now reformulate the acceptance probability (1.25), writing it in the explicit form needed to probabilistically solve the inverse problem (1.17) where we aim to generate independent samples from our target posterior $p(\mathbf{m}|\mathbf{d}_{obs})$.

$$\alpha(\mathbf{m}|\mathbf{m}') = min\left\{1, \frac{p(\mathbf{m}'|\mathbf{d}_{obs})q(\mathbf{m}|\mathbf{m}')}{p(\mathbf{m}|\mathbf{d}_{obs})q(\mathbf{m}'|\mathbf{m})}\right\} \tag{1.26}$$

substituting the posterior term given by eq.(1.20) we obtain

$$\alpha(\mathbf{m}|\mathbf{m}') = min\left\{1, \frac{p(\mathbf{d}_{obs}|\mathbf{m}')p(\mathbf{m}')q(\mathbf{m}|\mathbf{m}')}{p(\mathbf{d}_{obs}|\mathbf{m})p(\mathbf{m})q(\mathbf{m}'|\mathbf{m})}\right\} \tag{1.27}$$

which is known as *Metropolis-Hasting rule* (or M-H ratio). The transition kernel $q(\mathbf{m}'|\mathbf{m})$ defines a possible move of the Mc from the current model (state) $\mathbf{m}$ to a trial model $\mathbf{m}'$. The move has to be accepted, or rejected with probability α. For this reason the transition kernel is named *proposal* probability.

Metropolis updating scheme A special case of the M-H algorithm when the proposal is symmetrical $q(x, y) = q(y, x)$ is widely used for it's relatively simplicity of implementation. The symmetry of the proposal distribution leads to a simplification in the Metropolis-Hasting ratio (1.25) that takes the form:

$$\alpha(x, y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\} \tag{1.28}$$

The most typical way to implement a proposal scheme fitted out with a symmetrical proposal is to propose a trial model $y = x + \epsilon$ where ϵ is a random deviate normally or uniformly distributed around zero. The further consideration that since the prior distribution is not supposed to change between states the prior ratio is either one (when the proposed moves lies inside the prior-defined subspace) or zero (when it's outside) allows us to write the *Metropolis ratio* for the inverse problem (1.17) only through the likelihood ratio:

$$\alpha(\mathbf{m}|\mathbf{m}') = min\left\{1, \frac{p(\mathbf{d}_{obs}|\mathbf{m}')}{p(\mathbf{d}_{obs}|\mathbf{m})}\right\} \tag{1.29}$$

Metropolis updating scheme is summarized in Algorithm 1 using pseudocode.

Algorithm 1 METROPOLIS updating scheme

```
initialize m
for n = 1 : niter do
    propose m' = m + ε, where ε ~ N(0, σ)
    compute α = p(d_obs|m')/p(d_obs|m)
    generate u ~ U(0, 1)
    if u < α then
        accept m' = m
    else reject m'
    end if
end for
```

1.7.3 Transdimensional McMC

Metropolis-Hastings algorithm use has been generalized by Green (1995) who introduced the *reversible jump algorithm* as an extension of M-H to cases where the proposal distribution allows for transitions not only between models in the same state space, but also between state spaces of different dimensions. In this work the term "Transdimensional" will be used to refer to McMC implementations that allow for dimension-changing proposals. Introducing the index k to explicitly indicate the dimension of a space state (or model) the Metropolis-Hastings ratio (1.27) takes the form:

$$\alpha(\mathbf{m}, k|\mathbf{m}, \mathbf{k}') = min\left\{1, \frac{p(\mathbf{d}_{obs}|\mathbf{m}', k')p(\mathbf{m}', k')q(\mathbf{m}, k|\mathbf{m}', k')}{p(\mathbf{d}_{obs}|\mathbf{m}, k)p(\mathbf{m}, k)q(\mathbf{m}', k'|\mathbf{m}, k)} \cdot |\mathbf{J}|\right\} \tag{1.30}$$

Where $|\mathbf{J}|$ is the determinant of the Jacobian matrix of the transformation between models $\mathbf{m}$ and $\mathbf{m}'$. The type of transdimensional algorithm we will implement and utilize in Chapter.2 belongs to a special sub-family of the reversible jump algorithm where the jumps between dimensions are allowed to add or remove only one single variable (known as *birth-death* McMC (Green et al., 2003), and where the Jacobian term simplifies to $|\mathbf{J}| = 1$ (Sambridge et al., 2006).

1.7.4 The likelihood function

In the Bayesian framework different models need to be compared in the sampling process and for each of them the degree of fit to the data must be evaluated. the likelihood function $p(\mathbf{d_{obs}}|\mathbf{m})$ is a measure that quantifies how well a model $\mathbf{m}$ is able to reproduce a set of observed data $\mathbf{d_{obs}}$. In the case of seismic tomography, if we assume our data to be affected by experimental uncertainties, estimated by $\boldsymbol{\sigma}$, than one expression for the likelihood could be given through the L_2 misfit function:

$$\Phi(\mathbf{m}) = \left\| \frac{\mathbf{g}(\mathbf{m}) - \mathbf{d}^{obs}}{\boldsymbol{\sigma}} \right\|^2 \tag{1.31}$$

which gives a likelihood in the form:

$$\begin{aligned} p(\mathbf{d}_{obs}|\mathbf{m}) &= k \cdot \exp\left(-\frac{\Phi(\mathbf{m})}{2}\right) \\ &= k \cdot \exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\frac{g_i(\mathbf{m}) - d_i^{obs}}{\sigma_i}\right)^2\right] \end{aligned} \tag{1.32}$$

where $\mathbf{g}(\mathbf{m})$ is the data vector computed from the model $\mathbf{m}$ and σ_i are the estimated uncertainty on the data and n the dimension of the data vector (i.e. number of traveltimes). In this study we assumed the data uncertainty to be offset-dependent with values estimated through a linear interpolation between σ_{min} and σ_{max}, respectively the error that affects a minimum-offset and a maximum-offset travel time. k is a normalizing factor in the form $1/\sqrt{2\pi}\prod_{i=1}^{n}\sigma_i$ that is computationally irrelevant since, in the implementation of our McMC algorithms, likelihoods are always compared trough ratios.
In this study to monitor the time evolution of the level of data fit for the models within chains, a *nomalized misfit* function will be utilized:

$$M(\mathbf{m}) = \frac{1}{2(n-1)} \sum_{i=1}^{n} \left(g_i(\mathbf{m}) - d_i^{obs}\right)^2 \tag{1.33}$$

In this way one can observe how to a maximization of the likelihood function corresponds a minimization of the misfit. Practical use of $M(\mathbf{m})$ will be presented in the following chapters (e.g. Sec. 2.3.3).

1.7.5 Analyzing the esemble properties

Solving an inverse problem corresponds to infer the value of some parameter using observations (Tarantola, 2006). In a Bayesian framework this actualizes in the use of statistical inference on posterior distributions to falsify possible solutions, more than in the search of a single model. However it's a useful practice to seek a single solution out of the ensemble to be used for comparison with conventional linearized methods or for interpretation or simply for interpretation.
In principle the expectation of any function $h(\mathbf{m})$ of the model can be evaluated by means of the central limit (1.14) and ergodic (1.13) theorems as :

$$E_p[h(\mathbf{m})] = \int h(\mathbf{m}) p(\mathbf{m}|\mathbf{d}_{obs}) d\mathbf{m} \tag{1.34}$$

Note that the formula for the expectation given by the ergodic theorem reported in eq.(1.34) makes explicit use of the inverse-problem notation instead of the more general formulation of eq.(1.13).
A typical choice for the function to evaluate is that of a simple arithmetic mean, in order to extract a mean model as a spatial average of the posterior distribution the average of the velocity value assumed by each model parameter is computed from the posterior distribution as:

$$E_p[h(\mathbf{m})] = \frac{1}{M} \sum_{i=1}^{M} h(\mathbf{m_i}) \tag{1.35}$$

where h($\mathbf{m}$) is the model itself and M is the number of the models sampled and saved in the Markov chain.
Other possible reference solutions could be extracted from the posterior such as the *best* model, characterized by the lowest likelihood value (n.b. log-likelihood is being used in this study) or the *modal* model characterized by the maximum posterior value. Modal and mean model are supposed to correspond in case of unimodal, gaussian distributed posteriors. The mode could be a choice of particular interest while seeking a reference solution out of multimodal posterior distributions, being insensitive to outliers the mode is indeed not influenced by local minima.

Chapter 2

Transdimensional McMC

2.1 Reversible jump McMC

"Finding ways of sampling from trans-dimensional posteriors has been an active area of research in statistics culminating with the breakthrough papers of Geyer and Møller (1994) and Green (1995). The latter introduced what is became known as the reversible jump Markov chain Monte Carlo (rj-McMC) algorithm. This extended the familiar McMC method for sampling a fixed dimensional space into one for a general trans-dimensional problem." (Sambridge et al., 2006).

2.2 Method: Bayesian traveltime tomography

The standard deterministic way of approaching the tomographic inverse problem consists in the minimization of a target function by means of iterative methods and linearization. Bayesian theory on the other hand is a fully non-linear strategy where every information is regarded in terms of probability densities. The information content coming from the data is combined with the available *a priori* information in order to infer the *a posteriori* probability density through Bayes Theorem (see eq.1.19). The posterior probability density joins in this way all the information we may have on one problem, both from measurements and *a priori* information, and allows to display all the possible values that a parameter can take, together with their respective probability.

2.2.1 Prior distributions

Any knowledge on the model we have before the inversion process takes place, which can be expressed through a probability distribution, should be accounted for in the prior distribution $p(\mathbf{m})$. Our prior has been defined with ranges of

uniform probabilities, both for velocity and number of parameters. To represent our prior in terms of a probability distribution $p(\mathbf{m})$ we must consider that in the sampling process velocities and number of inversion nodes are independent, so the prior can be written as:

$$\begin{aligned} p(\mathbf{m}) &= p(\mathbf{m}|\mathbf{n})p(\mathbf{n}) \\ &= p(\mathbf{n}|n)p(\mathbf{v}|n)p(\mathbf{n}) \end{aligned} \tag{2.1}$$

Let us consider separately each component:
$p(\mathbf{n}|n)$ is the prior on the position of the inversion nodes. Assuming a maximum number of N parameters whose possible positions are defined, the probability to sample a model with n nodes is expressed through the binomial coefficient:

$$p(\mathbf{n}|n) = \binom{N}{n}^{-1} = \frac{n!(N-n)!}{N!} \tag{2.2}$$

$p(\mathbf{v}|n)$ is the prior on the velocity, constrained by the allowed velocity range $\mathcal{V} = \{v_i \in \mathbb{R} | v_{min} < v_i \leq v_{max}\}$:

$$p(v_i|n) = \begin{cases} 1/(\Delta v) & \text{for } v_i \in \mathcal{V} \\ 0 & \text{otherwise,} \end{cases}$$

where $\Delta v = v_{max} - v_{min}$. For both the models treated in this chapter $\mathcal{V} =]0.3, 8.0]$ Km/s. Considering that the velocity v_i is independent for every node, the distribution becomes:

$$p(\mathbf{v}|n) = \prod_i^n p(v_i|n) \tag{2.3}$$

$p(\mathbf{n})$ is the prior distribution of the number of parameters, namely the probability of a model to have n parameters, given simply by:

$$p(\mathbf{n}) = \begin{cases} 1/(\Delta n) & \text{for } n \in \mathcal{N} \\ 0 & \text{otherwise} \end{cases} \tag{2.4}$$

with $\Delta n = n_{max} - n_{min}$ defined in the set $\mathcal{N} = \{n \in \mathbb{N} | n_{min} \leq n \leq n_{max}\}$. For this synthetic model n_{max} was set to 110, while for the Salzach model the maximum number of nodes allowed is 60. Since at least one parameter is needed $n_{min} = 1$. Now the different terms (eq. 2.2, 2.3 and 2.4) of the prior (eq. 2.1) can be recombined to finally obtain the *a priori* distribution in the form:

$$p(\mathbf{m}) = \begin{cases} \dfrac{n!(N-n)!}{N!(\Delta v)^n \Delta n} & \text{for } v \in \mathcal{V} \text{ and } n \in \mathcal{N} \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

2.2.2 Proposals: how to move between models.

Proposal distributions give a statistical description of the probabilities to propose a move to a specific state in the model space, given the actual position. The probability $q(\mathbf{m}'|\mathbf{m})$ to move from model $\mathbf{m}$ to $\mathbf{m}'$ is determined by the proposal scheme used, namely by the kind of perturbations applied to a current model to obtain a new trial model. In our transdimensional algorithm two different perturbation kinds are in use. At every iteration of the Markov chain we decide with a uniform probability (user-defined) the kind of perturbation to be performed:

1) Velocity perturbation: A node is randomly chosen from the set of inverse parameters and its velocity is perturbed according to a Gaussian probability density as follows:

$$v_i' = v_i + n \cdot \sigma \tag{2.6}$$

where $n \in N(0,1)$ is a normally distributed random and σ is the standard deviation of the proposal. Note that σ does not correspond with the data uncertainty in equations 1.31 and 1.33 despite the use of the same symbol. For these fixed-dimension perturbations the proposal distribution is:

$$q(v_i'|v_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(v_i' - v_i)^2}{2\sigma^2}\right\} \tag{2.7}$$

eq.2.7 represents the probability density for a proposed velocity value v_i': normally distributed, with mean value v_i and standard deviation σ. It is trivial to observe that since $q(v_i'|v_i) = q(v_i|v_i')$ such a perturbation is symmetrical, the probabilities of moving from $\mathbf{m}$ to $\mathbf{m}'$ and backwards are equal. In this case the proposal ratio simplifies to unity:

$$\frac{q(\mathbf{m}|\mathbf{m}')}{q(\mathbf{m}'|\mathbf{m})} = \frac{q(v_i|v_i')}{q(v_i'|v_i)} = 1 \tag{2.8}$$

2) Node status perturbation: it's a transdimensional step where one randomly decides wether to remove one model parameter from the inversion or to add a new one, the choice is taken with equal probability. Removing and adding parameters is done respectively switching the status of a node from *inverted* to *interpolated* and vice versa.

Node Birth: In this perturbation step the status of one of the interpolated nodes is switched to inverted, a new model parameter is created with probability :

$$q(\mathbf{n}'_{n+1}|\mathbf{m}) = \frac{1}{N-n} \tag{2.9}$$

where N is the total number of nodes, and n is the number of inverse nodes (number of model parameters). The probability of generating a new velocity for the new node is independent from the number of model parameters and is:

$$q(\mathbf{v}'_{n+1}|\mathbf{m}) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(v'_i - v_i)^2}{2\sigma^2}\right\} \tag{2.10}$$

Node Death: this is the inverse birth step where we mode from a model with $n+1$ to another with n parameters. The probability of a node status to be switched from *inverted* to *interpolated* is:

$$q(\mathbf{n}_n|\mathbf{m}') = \frac{1}{n+1} \tag{2.11}$$

and since when a parameter is removed its velocity value follows the same fate:

$$q(\mathbf{v}_n|\mathbf{m}') = 1 \tag{2.12}$$

Having observed that generating new model parameters and assigning velocity values to these parameters (and vice versa) are independent events we can write the proposal ratio for transdimensional steps as:

$$\frac{q(\mathbf{m}|\mathbf{m}')}{q(\mathbf{m}'|\mathbf{m})} = \frac{q(\mathbf{n}_n|\mathbf{m}')}{q(\mathbf{n}'_{n+1}|\mathbf{m})} \cdot \frac{q(\mathbf{v}_n|\mathbf{m}')}{q(\mathbf{v}'_{n+1}|\mathbf{m})} \tag{2.13}$$

Substituting the terms in equations (2.9), (2.10), (2.11) and (2.12) into eq.(2.13) we obtain:

$$\frac{q(\mathbf{m}|\mathbf{m}')}{q(\mathbf{m}'|\mathbf{m})} = \begin{cases} \dfrac{\sigma\sqrt{2\pi}(N-n)}{n-1} \exp\left\{\dfrac{(v'_i - v_i)^2}{2\sigma^2}\right\} & \textbf{birth} \\ \dfrac{n}{\sigma\sqrt{2\pi}(N-n+1)} \exp\left\{-\dfrac{(v'_i - v_i)^2}{2\sigma^2}\right\} & \textbf{death} \end{cases} \tag{2.14}$$

The standard deviation σ has been scaled after recursive tests in order to obtain an acceptance ratio reasonably close to 23.4%, value that should ensure good mixing properties (Roberts et al., 1997). The perturbation variance in use in this chapter corresponds to $\sigma = 0.05$ Km/s.

2.2.3 Transdimensional acceptance ratios

The reversible-jump algorithm (Green, 1995) extends the abilities of the Metropolis-Hastings algorithm, allowing to apply it to cases where the number of model parameters can change between successive states and the dimension of the model space itself is a variable in the sampling process. Recalling the expression for the M-H acceptance ratio (1.30) we gave in the first chapter, and dropping for sake of simplicity the cardinality of the state space k, we can write the probability of a proposed transdimensional move to be accepted as:

$$\begin{aligned} \alpha(\mathbf{m}|\mathbf{m}') &= min\left\{1, \frac{p(\mathbf{m}'|\mathbf{d}_{obs})}{p(\mathbf{m}|\mathbf{d}_{obs})}\right\} \\ &= min\left\{1, \frac{p(\mathbf{m}')}{p(\mathbf{m})} \cdot \frac{p(\mathbf{d}_{obs}|\mathbf{m}')}{p(\mathbf{d}_{obs}|\mathbf{m})} \cdot \frac{q(\mathbf{m}|\mathbf{m}')}{q(\mathbf{m}'|\mathbf{m})}\right\} \end{aligned} \tag{2.15}$$

With this formulation we are sure that the marginal distribution of a Markov chain will eventually converge to the posterior distribution, and we can therefore compute the acceptance ratios for velocity, birth and death perturbations substituting into eq.(2.15) the expressions for prior, likelihood and proposal ratios obtaining:

Velocity step

$$\alpha(\mathbf{m}|\mathbf{m}') = min\left[1, \exp\left\{-\frac{\Phi(\mathbf{m}') - \Phi(\mathbf{m})}{2}\right\}\right] \tag{2.16}$$

Birth step

$$\alpha(\mathbf{m}|\mathbf{m}') = min\left[1, \frac{\sigma\sqrt{2\pi}}{\Delta v} \exp\left\{\frac{(v'-v)^2}{2\sigma^2} - \frac{\Phi(\mathbf{m}') - \Phi(\mathbf{m})}{2}\right\}\right] \tag{2.17}$$

Death step

$$\alpha(\mathbf{m}|\mathbf{m}') = min\left[1, \frac{\Delta v}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(v'-v)^2}{2\sigma^2} - \frac{\Phi(\mathbf{m}') - \Phi(\mathbf{m})}{2}\right\}\right] \tag{2.18}$$

where the above probabilities become zero in case number of parameters and velocity values are outside the priors. Let us briefly point out that in the acceptance rates the terms related to the prior distributions will tend to push the sampling process towards simpler models with less parameters while the likelihood term will tend to favor better fitting models. This property of Bayesian inference naturally prevents over parametrization of models in the attempt to achieve a better data fit, and is referred to as "natural parsimony".

2.2.4 Updating scheme

We implemented our transdimensional McMC algorithm as a module of the `simulr16` code. This allows to start off the inversion process with a deterministic inversion and to initiate the Markov chain setting the DLS solution as a starting model. A random initialization of the chain is of course possible, this choice naturally influences the duration of the *burn-in phase* (see Sec.2.3.3), no other difference is to be found in the sampled ensemble since the stability distribution is independent from the initial state. Our algorithm can be schematized as follows (Fig.2.1) :

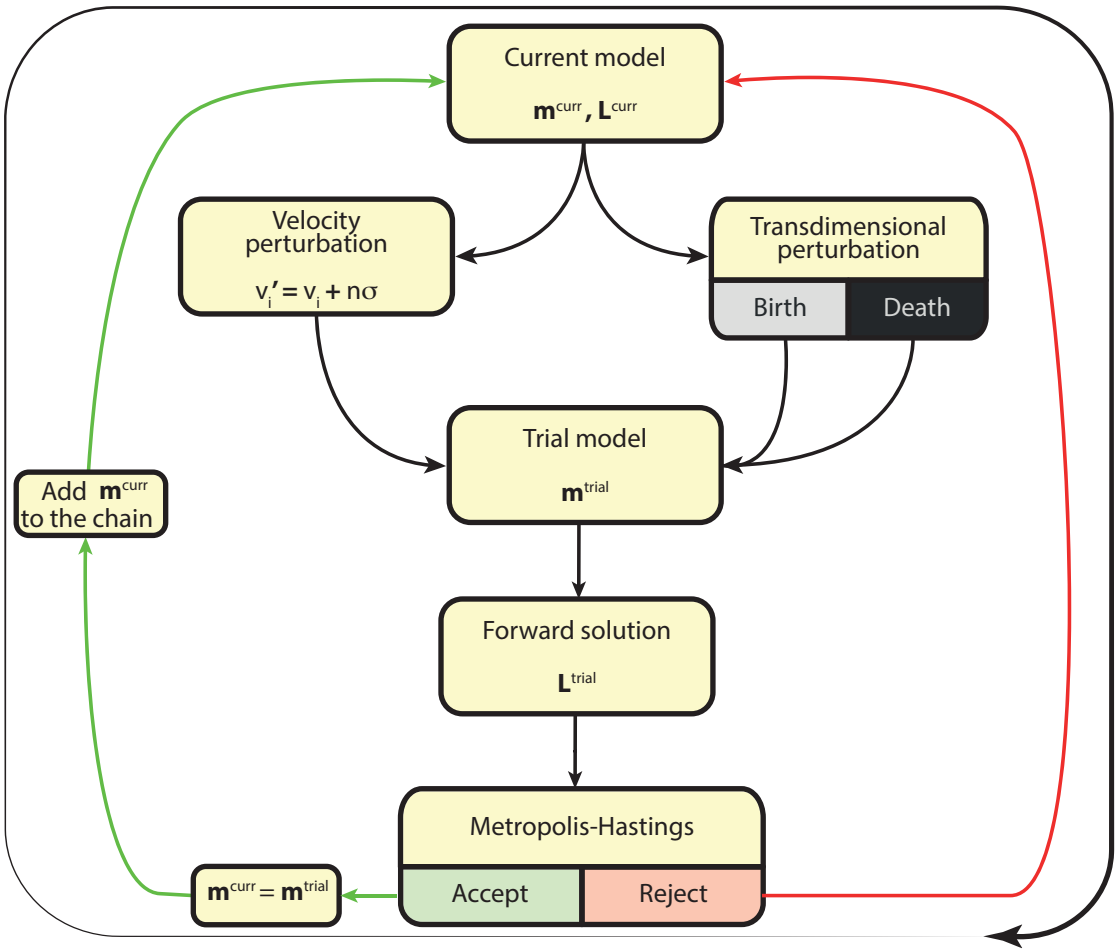


Figure 2.1: *Iterative updating scheme of our transdimensional McMC algorithm.*

2.3 Synthetic dataset

Our algorithm has been first tested on the synthetic model in Fig.2.2.a. The profile is 120 meters long and on its surface 23 sources and 23 receivers have been located with a distance of 5 meters. The synthetic travel times have been computed using the FAST algorithm (Zelt and Barton, 1998), gaussian random noise has been added to the data using a standard deviation of 5% of the noiseless travel time. The structure presents two layers (L1, L2) shaped to resemble the structure of a valley with a sedimentary filling with a constant vertical velocity gradient and a smooth interface. The upper layer is characterized by two low-velocity anomalies: a shallow one located on the surface between 25 and 40 meters extending in depth to 4 meters (A1), characterized by a p-velocity of 0.6 km/s. The second anomaly with a velocity of 1.0 km/s is located between 4 and 10 m in depth and extends between profile coordinate 70 and 80m (A2). The ray geometry corresponding to the sources 1,12,23 in our synthetic model is depicted in Fig.2.2.b.

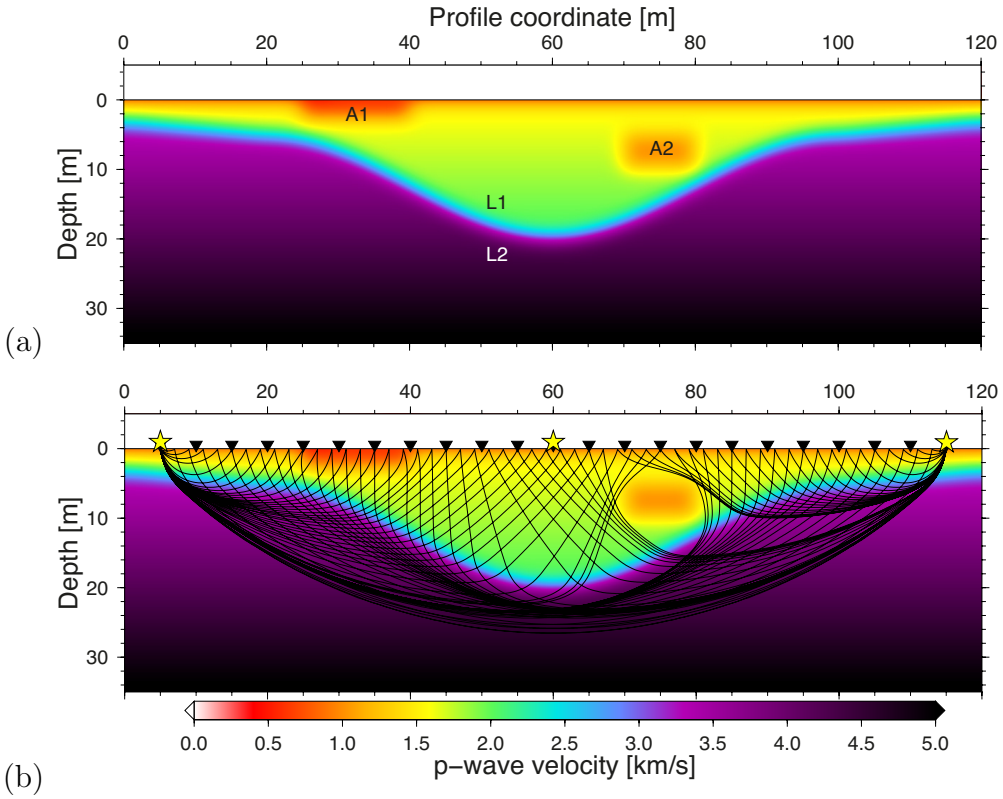


Figure 2.2: *The synthetic model used in this chapter: (a) the main structural features are here tagged as A1,A2 (low velocity anomalies) L1,L2 (layers). (b) Receiver locations (triangles) and ray paths for three of the sources (stars).*

2.3.1 Multiple Parallel Markov Chains

High-dimensional, non-linear inverse problems can be really computer-intensive. The models that we are presenting and inverting in this paper can easily require 2-4 weeks-long runtimes on a desktop computer (Intel Core i7-2600 Processor, at 3.40 GHz) in order to sample a satisfying number of models. It's therefore obvious that solutions to speed up the sampling process are highly desirable. Parallel computing technology is nowadays easily accessible thanks to the wide diffusion of multiple-cores, multiple-CPUs architectures on almost every consumer workstation. Parallelizing our algorithm seemed, hence, a straightforward solution to the need of a speedup in the sampling process. There are two different approaches in the parallelizations of McMC algorithms: parallelizing a single Markov Chain process, or running multiple chains in parallel. Given the intrinsic serial nature of McMC simulations, the most immediate of the approaches is therefore to generate a separate Markov Chain on different CPUs/cores and then appropriately combine their results (Brockwell, 2006). The main advantage of this "multiple chains" approach is that almost no extra coding needs to be done. One main issue has to be taken into account in this parallel computing environment: the burn-in. A portion of the time spent by every chain is "wasted" producing samples that must be discarded, and since a burn-in period must be spent on each of the multiple chains, the final speed-up function will be less then linear (Rosenthal, 2000), as shown in Fig.2.3.

$$Speedup(N) = \frac{m+b}{\frac{m}{N}+b} \tag{2.19}$$

In eq.(2.19) b is the number of the burn-in models, m are the post-burn-in models, and N the number of available CPUs.

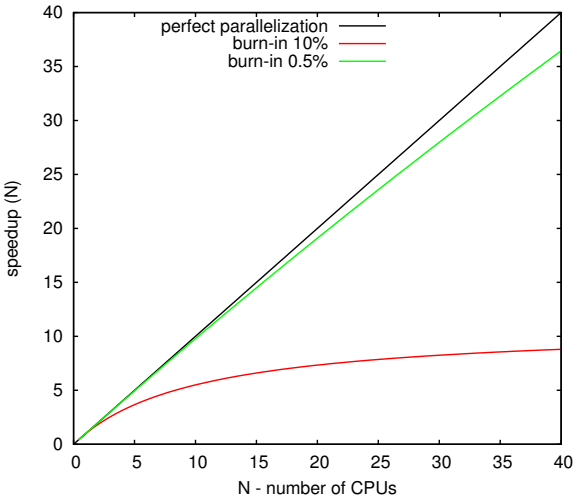


Figure 2.3: *Speedup relations for a multiple chains approach showing the theoretical linear speedup (black) in case of no burn-in (perfect parallelization), the curve (red) for an hypothetical case where the initial* 10% *of the models is discarded and the curve (green) for the* 0.5%*-burn-in we obtained in the inversion of the Salzach dataset (see section 2.4)*

2.3.2 Staggered grids

A useful approach that promises a reduction of the indetermination in tomographic inversions is the use of *staggered grids*. The procedure consists of separate inversions each of which is conducted with a differently-positioned inversion grid. The change of position of each grid for the different inversions is determined by shifts of defined length in opposite directions. For the synthetic dataset we adapted a conservative strategy applying one single stagger-step (Fig.2.4), however the direction, magnitude and number of stagger-shifts are parameters that could be adapted to the quantity of computational resources available and to each specific case-study. The final velocity field is obtained space-averaging the solutions obtained with the single inversions. In a deterministic framework this approach brings two main advantages both deriving from the use of a fairly coarse parametrization in the single inversions: the null-space energy is restrained and the computational costs are limited if compared with an equivalent fine, and probably over parametrized single-grid inversion (Böhm et al., 2000).

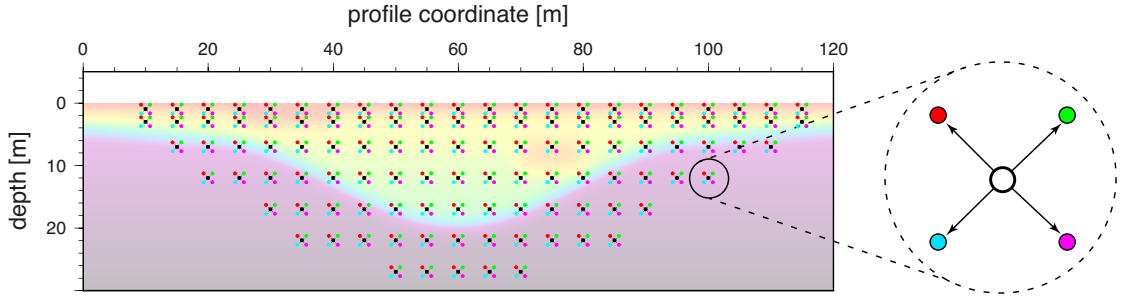


Figure 2.4: *Staggered grids: the original grid is shifted in four directions.*

In a probabilistic framework the practice of running and joining multiple independent Markov chains can be seamlessly flanked by staggered-grids. In our case this choice appears particularly appropriate because of a limitation related to the inverse parametrization strategy native in the `simul` code family. The pre-defined node positions are determining a fixed set of inversion-nodes that provides a useful and rational prior on the number of parameters but at the same time sets a limit to the possible positions of the nodes. This is somehow a suboptimal occurrence for trans-dimensional Markov chains that have in this way a limitation in their ability to adapt the parametrization to the data. The approach of applying a finer node parametrization is to be discouraged, since this would lead to higher computational costs while no substantial improvements would be seen in the posterior distributions obtained. In order to justify this statement, a small synthetic test was performed, visually comparing the probability density functions obtained from two Markov chains that sampled model spaces with different cardinality (i.e. same

synthetic model, with respectively a coarse and a fine parametrization). The PDF in Fig.2.5.a displays the probability distribution of velocity values at the depths where nodes are present, and it was obtained after $\approx 10^5$ iterations. Figure 2.5.b displays the conditional distribution obtained after the same number of iterations utilizing a 5-times finer node density in depth, thus a number of inverse parameters 5 times higher. We can observe that the distribution doesn't appear as smooth as the one obtained with a coarse parametrization, this could be interpreted as a sign of non-convergence: the chain has in fact to sample a model space with a higher cardinality and needs more time to reach convergence. Letting the sampler run for longer time we can see that eventually the conditional after $\approx 10^6$ iterations appears to be much smoother (Fig.2.5.c) thus the chain is likely to have reached convergence. The two approaches lead to distributions that at the same depths have compatible values, on the other hand the sampler needs approximatively 10 times more iterations in the fine-parametrized case resulting in suboptimal use of computational resources. A staggered-grid approach seems to be therefore a better strategy that allows to join at the same time the need for higher spatial resolution and for limited runtimes.

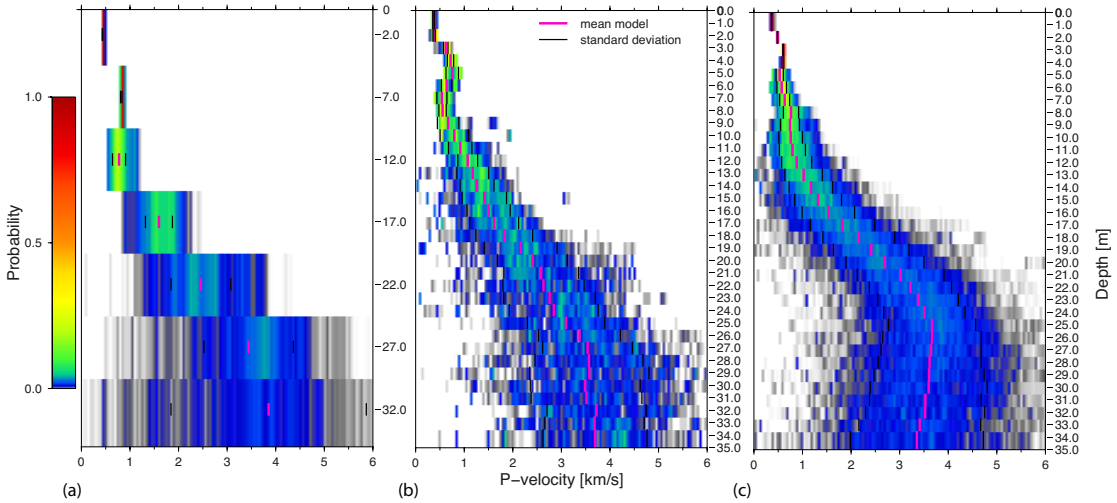


Figure 2.5: *Conditional probability density functions comparison: the same model is parametrized with a coarse node parametrization (a) and with a 5 times finer node spacing (b) and (c). The probability distributions are plotted at the same profile position only at depths where a node is present. The PDF in (a) and (b) are plotted after* 10^5 *iterations, while in (c) after* 10^6*.*

For the inversion of our synthetic data at every staggered-grid position multiple chains are run in parallel and then joined. Plotting the temporal evolution of the normalized misfit values (eq. 1.33) of the accepted models for the five chains (corresponding to the original parametrization, plus the four staggered positions)

we can visually assess that all the chains sample the model space providing models whose data-fit is comparable, with misfit values oscillating in the range 100-140 ms (Fig.2.6).

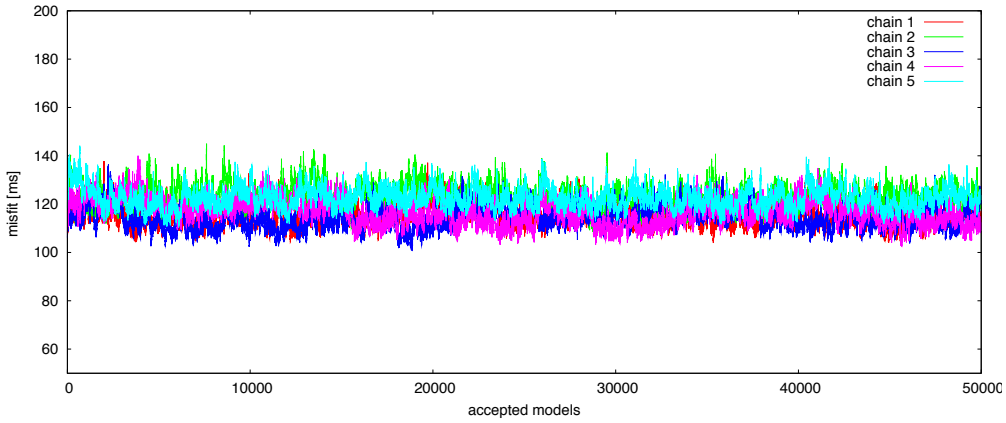


Figure 2.6: *Temporal evolution of the normalized misfits for five instances of staggered chains*

2.3.3 Burn-in and convergence estimation

The length of the burn-in period and its determination are crucial factors in the performance and quality of parallel Markov chain Monte Carlo algorithms. Since our McMC code only produces the chain of models and no statistical analysis is performed during the runtime, we could have visually analyzed the temporal evolution of the likelihood values and safely quantify the models to be discarded.

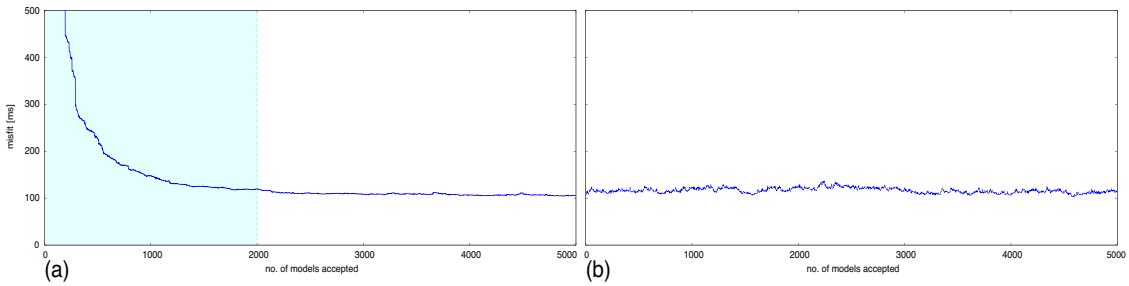


Figure 2.7: *Temporal evolution of normalized misfit values of the first 5000 models saved in two different Markov chains. (a) the chain initialized with a random model-state needs to have the first part of the models rejected (burn-in phase highlighted in the cyan box). (b) Having initialized the sampling process with the DLS solution no burn-in is necessary.*

As an alternative to this heuristic quantification, the choice of an appropriate starting point can also help to reduce or eliminate the burn-in time if the first model is chosen close to the target distribution (Fig.2.7). A convenient option we adopted is to initialize the sampling process with a model solution of a deterministic Damped Least Squares inversion.
Convergence assessment is a challenging task especially for transdimensional McMC applications where the parametrization is constantly being modified. In this situation it's not possible to simply monitor the velocity values of some specific nodes since they are subject to "birth-death" perturbations. A good practice that holds its validity in transdimensional application takes advantage of the CLT for Markov chains that under the reversibility condition grants asymptotic convergence to normality. Operatively one could simply monitor the time evolution of the likelihood (or also misfit) distribution (Fig.2.8): once out of the burn-in phase the PDF of the likelihood should asymptotically stabilize.

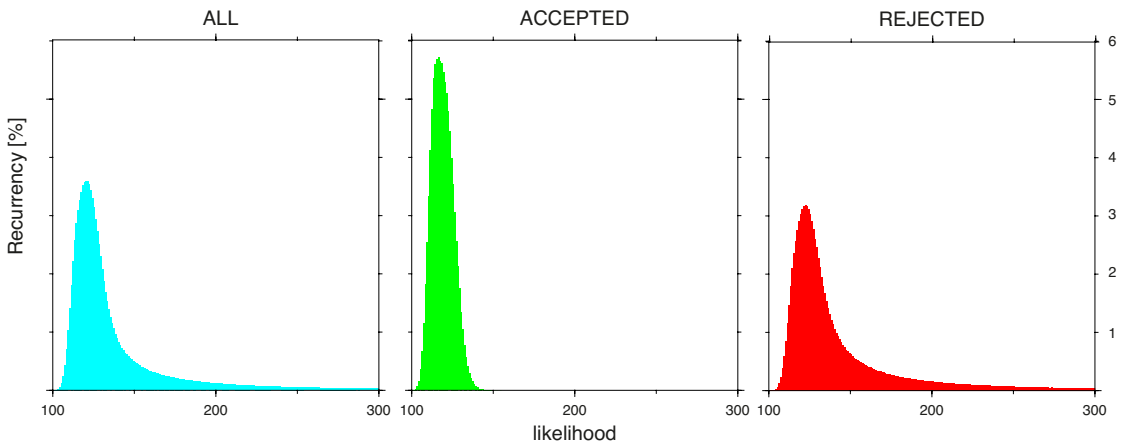


Figure 2.8: *Likelihood PDFs : probability distribution of the likelihood values of a Markov chain for (cyan) all models proposed, (green) accepted models, (red) rejected models.*

One also could monitor variance, mean and modal value of these distributions in order to obtain a quantitative tool to evaluate the convergence of a chain and to establish a stopping-criterion (i.e. when to stop the sampling process). Monitoring the number of inversion parameters during an inversion could be a possible convergence-tool still supported by the CLT, however we practically noted in this study that the dimensionality of the problem often tends to stabilize quickly at a stable value, this could mislead to asses as convergence what is actually a pseudo-convergence (sometimes referred as multimodality). A last simpler alternative is again a graphical analysis of the likelihood traces (Fig.2.6) which should permanently fluctuate around an average value.

2.3.4 Inversion

The synthetic dataset has been inverted with our transdimensional McMC algorithm. The sampling process, initialized with the deterministic solution of a DLS inversion (Fig.2.9), was let run for ≈ 34 days utilizing staggered grids and multiple chains. Each staggered-grid inversion was produced shifting the original grid of $2\sqrt{2}$ meters along diagonal directions (2 m horizontal and vertical shifts) as illustrated in the scheme of Fig.2.4. For every staggered position 4 parallel chains run independently. During a total (over the 20 total chains) of 14.19 Million iterations 3 million models were accepted as member of the chains, for an average acceptance rate of ≈ 21%. The stopping-criterion adopted aimed to the collection of 3 million models, value that by means of previous runs of the algorithm has been estimated as widely sufficient to avoid pseudo-convergence and to grant the *saturation* of the sampled posterior distribution.

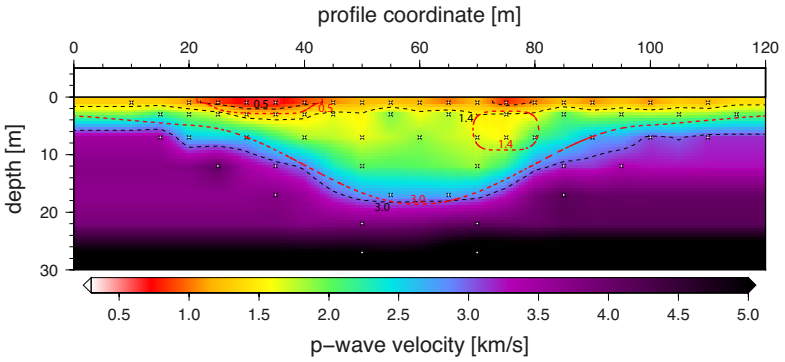


Figure 2.9: *Deterministic solution model used to initialize the Markov Chain. Isolines of the synthetic model are reported for comparison as a red dashed curve. Inversion nodes are marked with crosses.*

In order to ease the comparison with the reference synthetic model, its velocity contour lines are reported as a red dashed curve. The surface low velocity anomaly (A1), the middle one (A2), and the bedrock shape (interface L1/L2) are visualized with isolines, characterized respectively by 0.5, 1.4 and 3.0 km/s p-waves velocities. The inverse parametrization displayed in Fig.2.9 has been determined adapting the number and position of the inverse nodes with a procedure based on the Resolution Diagonal Elements (Fig. 2.10) as described by Bleibinhaus and Gebrande (2006).

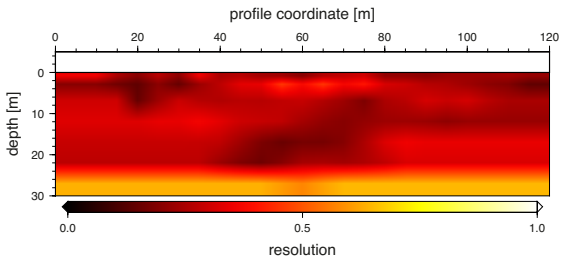


Figure 2.10: *RDE map: Resolution Diagonal Elements*

2.3.4.1 Posterior distributions PDF

Probability density functions (PDFs) or *posterior distributions* can be referred as the actual "solutions" of an inverse problem from a Bayesian perspective. The statistical information contained in the PDF provide all what is needed to infer the most important properties of the sampled model ensemble. Posterior distributions of the velocity field obtained with our multiple-staggered grid approach are shown in Fig.2.12 as vertical profiles at profile coordinates 35, 60 and 75 m in correspondence of the two low velocity anomalies (A1, A2) and at the centre of the profile. These PDFs are computed from the ensemble of sampled models and display all the velocity values assumed by each single inverse parameter, together with the relative probability (normalized to each depth).

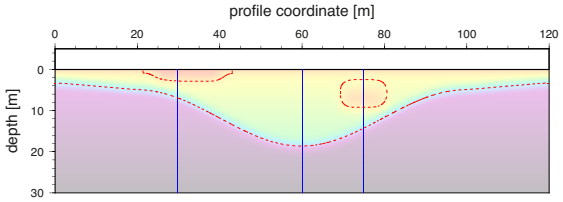
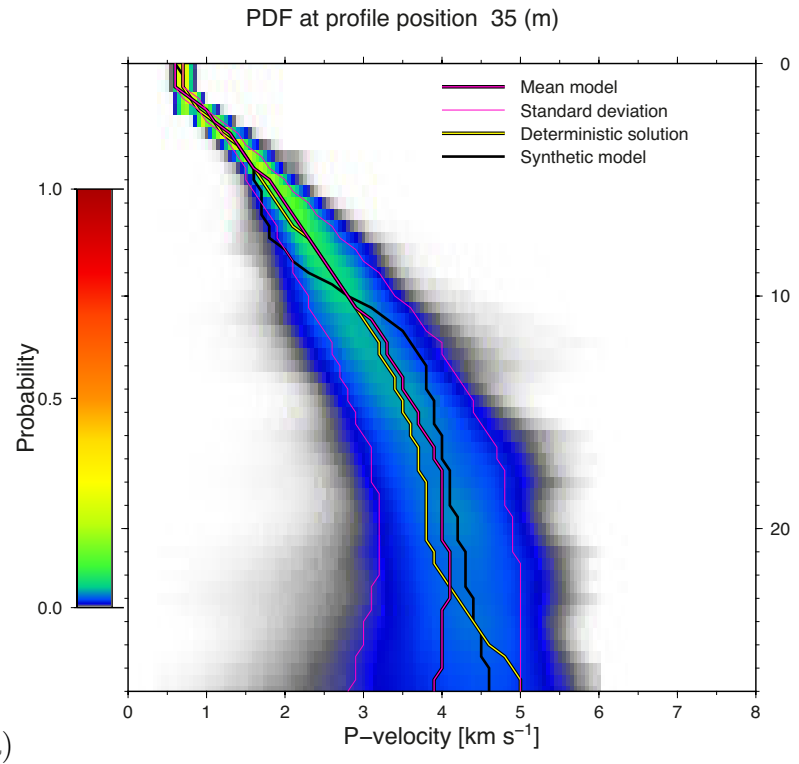


Figure 2.11: *Locations of the PDF vertical cross sections on the model: solid blue lines.*

Figure 2.12: *continues on next page*



(a)

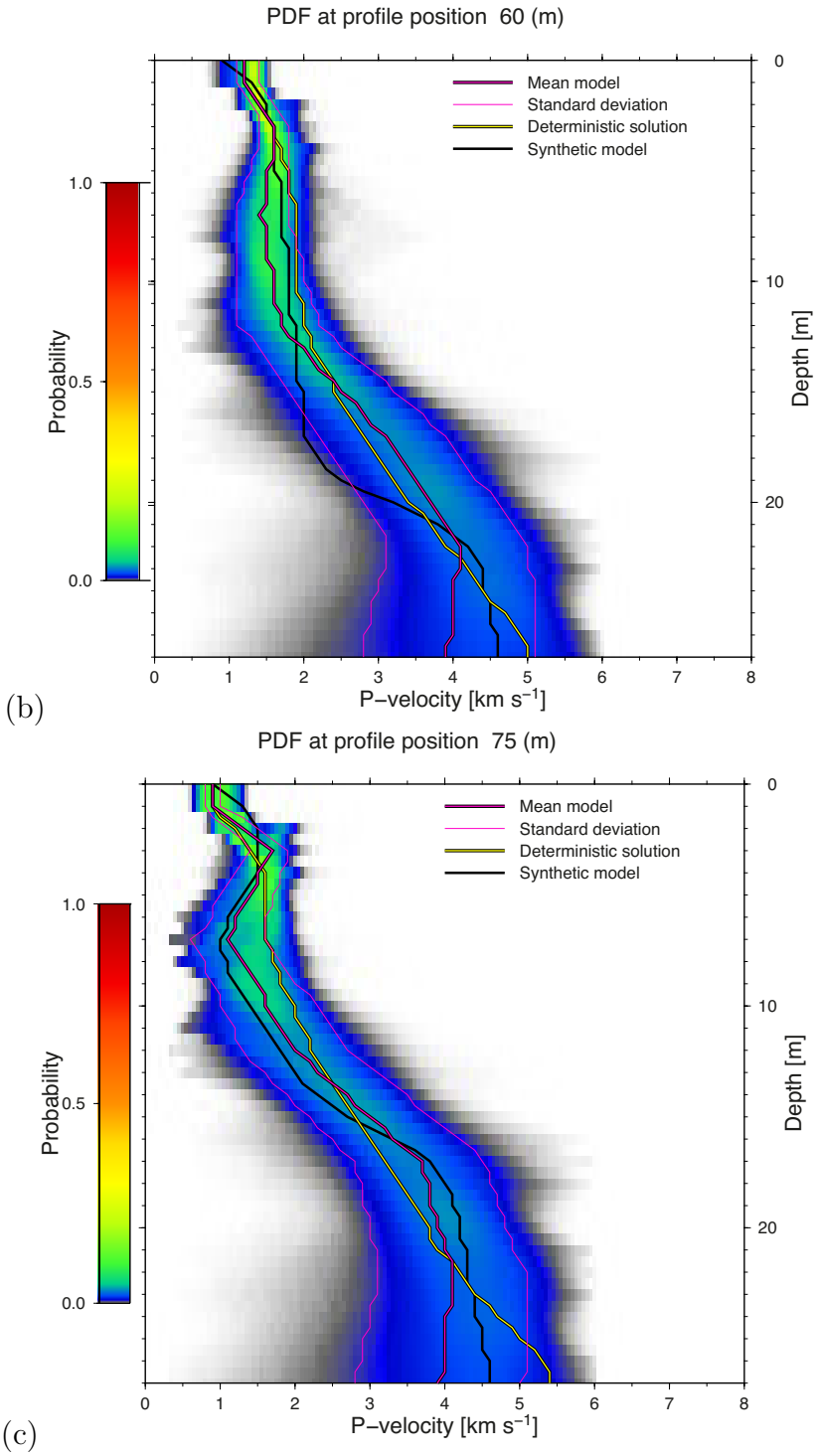


Figure 2.12: *Probability density functions (PDF): vertical cross sections at profile positions 30, 60 and 75 m. displaying the posterior velocity distribution as well as vertical profiles of the synthetic model, of the DLS solution, the mean posterior and its confidence interval delimited by one standard deviation.*

2.3.4.2 Mean and Modal solutions

The posterior information carried by the ensemble of sampled model was extracted in terms of mean and modal models as pointed out in Section 1.7.5. Mean models solutions were computed for both single- and multiple-staggered-chains.
A comparison presented in Fig.2.13 between the mean-solution maps obtained from a single chain and that obtained from multiple-staggered chains shows similar results: the low-velocity anomaly A1 is in both cases correctly reconstructed in shape and extenstion, while the synclinal shape representing the valley bedrock is recovered with a depth-discrepancy of a couple of meters from the synthetic value (see red-dashed line in Fig.2.13). The deeper anomaly A2 is partially recovered with the multiple-staggered method that provides a better matching velocity value while the single chain tends to slightly underestimate the velocities in the near surface and to smear the anomaly. In both cases a low-velocity artifact is to be noticed at the surface around profile meter 75 in correspondence of the actual position of the low-velocity anomaly.

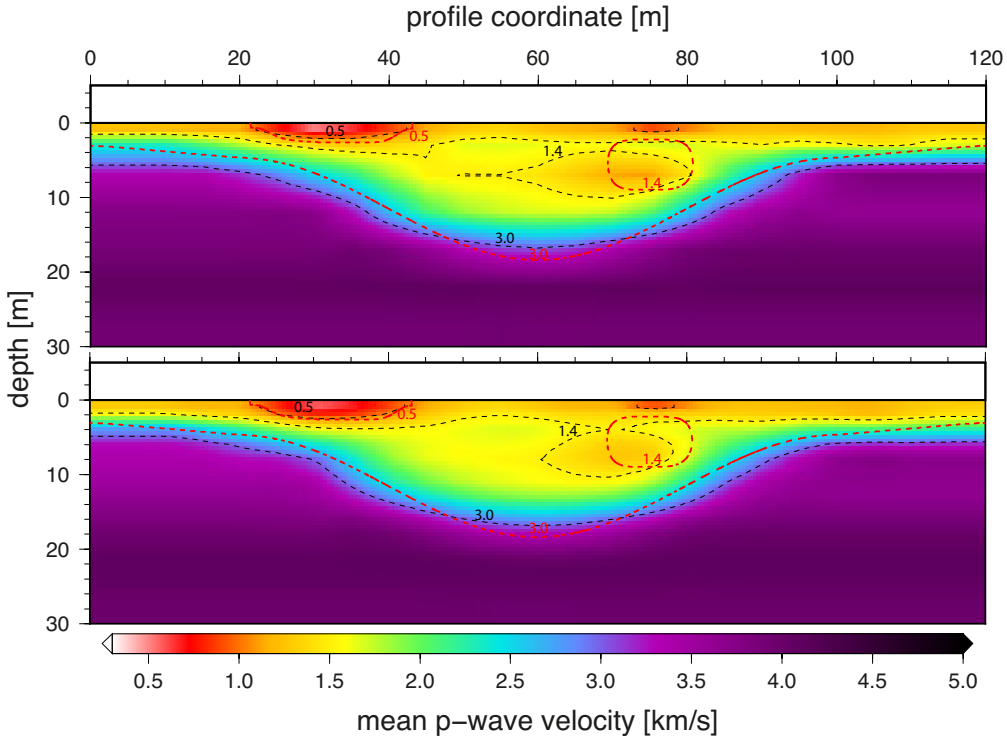


Figure 2.13: *Mean model solutions compared: single Markov chain (top) multiple staggered chains (bottom). The red-dashed isolines are relative to the synthetic model while the black ones to the mean solutions.*

Alternatively, or in addition to a mean-model solution, one could employ the *modal*

solution model obtained as the maximum of the posterior distribution. In case of skewed or multi-modal PDFs the analysis of the modal velocity map could provide information useful for the interpretation. In the modal-solution in Fig.(2.14) the deeper low-velocity anomaly appears less smeared, in addition, the high velocities in the bottom part of the model are closer to the synthetic values than the corresponding mean values.

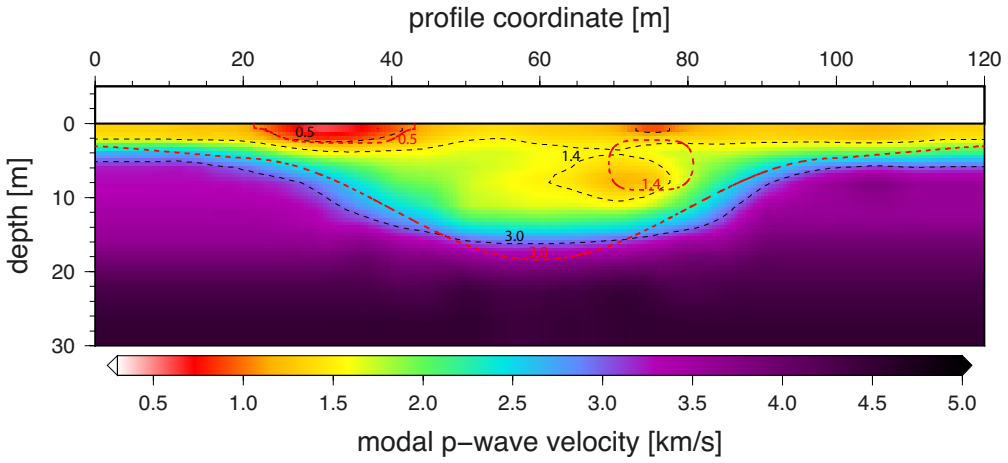


Figure 2.14: *Modal velocity model corresponding to the multiple-staggered Markov chain.*

2.3.4.3 Uncertainty map

A valuable tool in bayesian tomographic inversion is a quantitative error estimation. CLT and ergodic theorem once again provide the support to the computation of error maps through eq.(1.34) where the expectation is computed on $h(\mathbf{m})$ that here becomes the variance of the posterior:

$$E_p\left[h(\mathbf{m})\right] = \frac{1}{M}\sqrt{\sum_{i=1}^{M}\left(\mathbf{m}_i - \hat{\mathbf{m}}\right)^2} \tag{2.20}$$

As expected well-constrained areas are located on the surface of the model, while the bottom part has higher uncertainty (Fig.2.15). The depth dependency of the standard error can be also visualized on the posterior density functions (Fig.2.12). The relative standard error map (Fig.2.16) highlights the central area of the model as that where the larger relative discrepancies with the synthetic model are to be expected. This map is derived computing the ratio between the standard deviation and the mean model, normalized to 100.

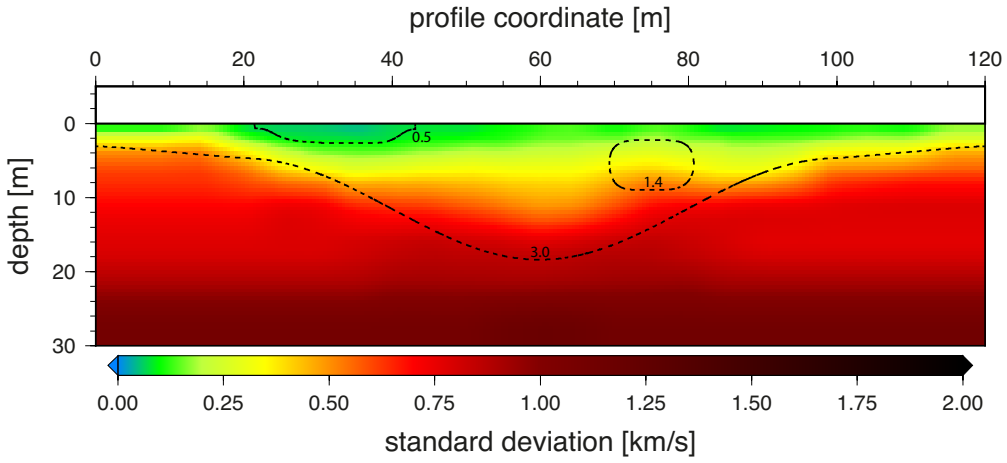


Figure 2.15: *Standard deviation map with contour lines of the synthetic structure.*

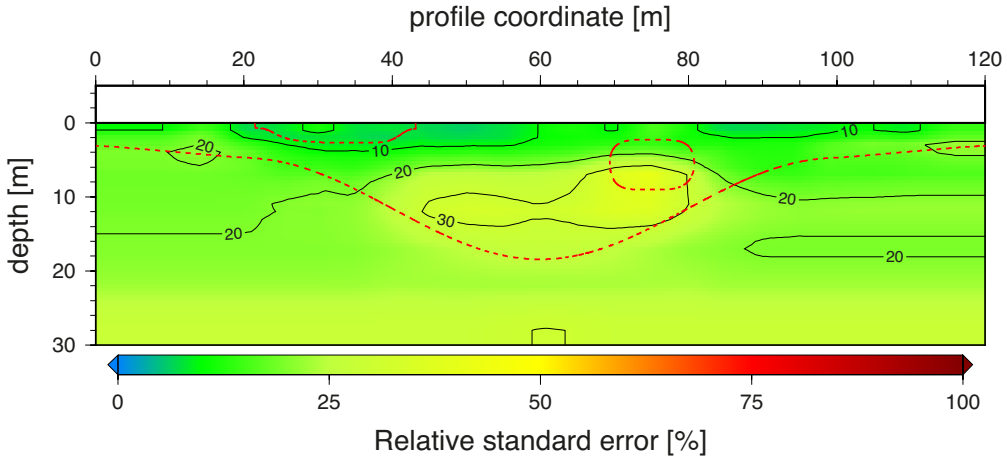


Figure 2.16: *Quantitative error maps: standard error (top) and relative error (bottom). The dashed contour lines display for comparison the synthetic structure.*

2.4 Salzach valley

The transdimensional McMC algorithm with multiple-staggered chains was tested on a seismic dataset acquired in the year 2009 with the purpose to image the shape and structure of the Salzach Valley near Zell-am-See, Austria. A seismic line was deployed perpendicularly to the Salzach river running across the valley and crossing the valley floor and part of the bedrock at its extremities (Fig.2.17).

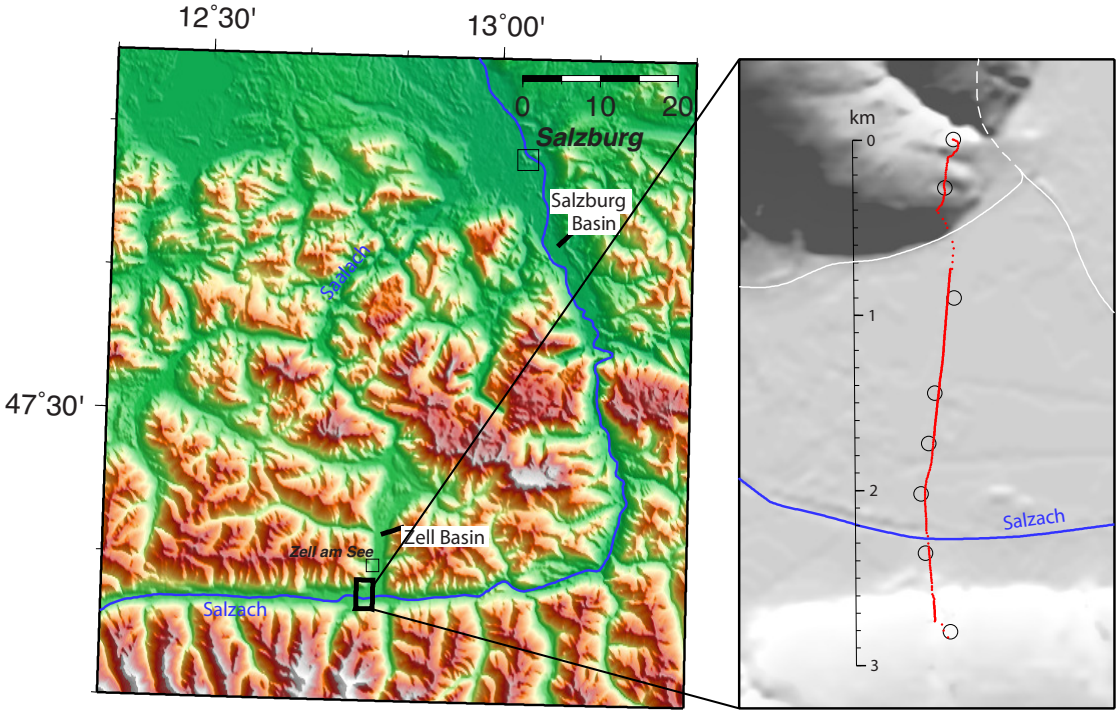
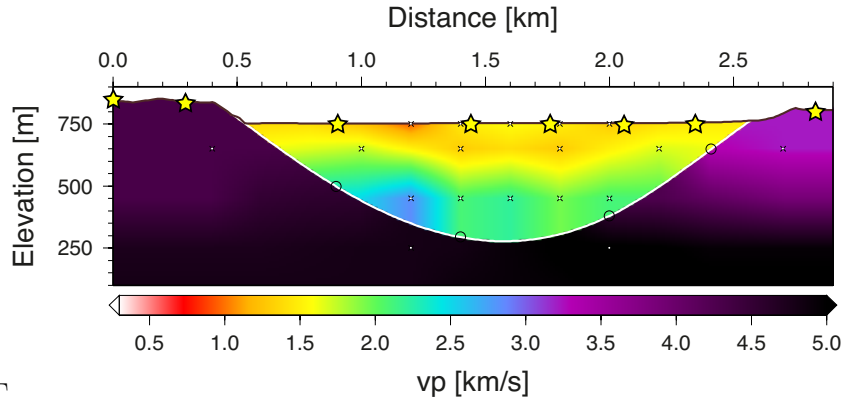
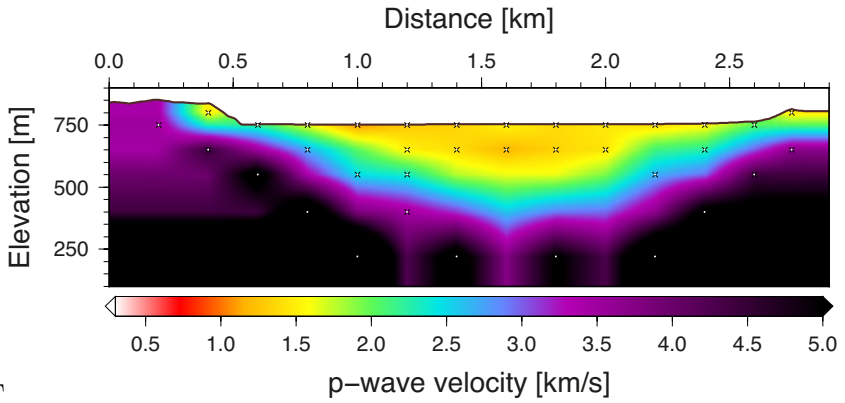


Figure 2.17: *Map of the Salzach river valley (left), the inset maps the investigation area. Seismic profile with shot locations (circles) and receivers (red line).*

The survey geometry counts eight explosive sources with an average spacing of 400 m and a fixed-spread line of 10 Hz vertical-component geophones, with 10 m spacing. First results from first-arrival traveltime tomography (FATT) were published by Bleibinhaus et al. (2010). In deep interpretation including refraction-reflection traveltime tomography (RRTT), full waveform inversion (FWI), radar and resistivity methods have been presented by Bleibinhaus and Hilberg (2012).
The inverse parametrization for RRTT is formed by a set of velocity and a reflector-depth nodes. The two-steps velocity interpolation performed to obtain a smooth velocity field starting from the coarse node-based parameterization is described in Bleibinhaus and Hilberg (2012).
The models of seismic velocity by Bleibinhaus and Hilberg (2012) show a quite



(a) RRTT



(b) FATT

Figure 2.18: *Deterministic DLS solutions obtained inverting refraction-reflection data (a) and first arrivals only (b). The RRTT solution of Bleibinhaus and Hilberg (2012) is reported on the left (a) together with the node parametrization used for the inversion (crosses), reflector parameters (circles) and the explosive sources (stars). On the right (b) the FATT solution used in this study as starting model for the McMC process.*

heterogeneous structure of the valley infill with no signs of layering, except for a shallow zone of high velocities down to ≈50 m depth in the centre of the valley (A in Fig.2.22.b) interpreted as an aquifer surrounded by regions of lower velocity (B) with a thickness of ≈150 m. The deeper part of the basin is characterized by a higher-velocity zone (C) more visible in the RRTT model with velocities between 2.5 and 3 km/s.

2.4.1 Transdimensional McMC inversion results

The Salzach dataset has been inverted with our transdimensional multiple-staggered McMC method: the staggering process counts two staggered steps in each diagonal

direction with a magnitude of 20 meters each for a total of eight shifts/positions plus the original one (Fig.2.19). At every staggered position we run three parallel independent Markov chains, whose models where joined before the stagger-averaging process.

Models accepted per chain	200000
Thinning	20
Models saved per chain	10000
Staggered positions	9
Multiple chains per position	3
Total models accepted	5400000
Total models Saved	270000

Table 2.4.1: *Inversion data recap*

The number of staggered steps and parallel instances are optimized on the quantity of the available cores and workstations. Each sampler was set to run until 10000 models were saved, collecting an ensemble of 270000 models corresponding to a set of 5.4 millions of accepted models thinned with a 20-model spacing.

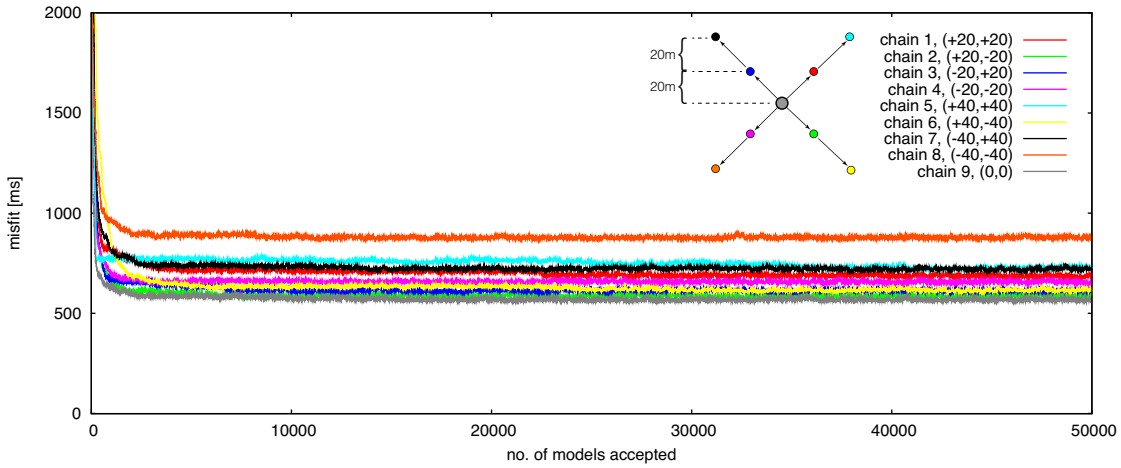
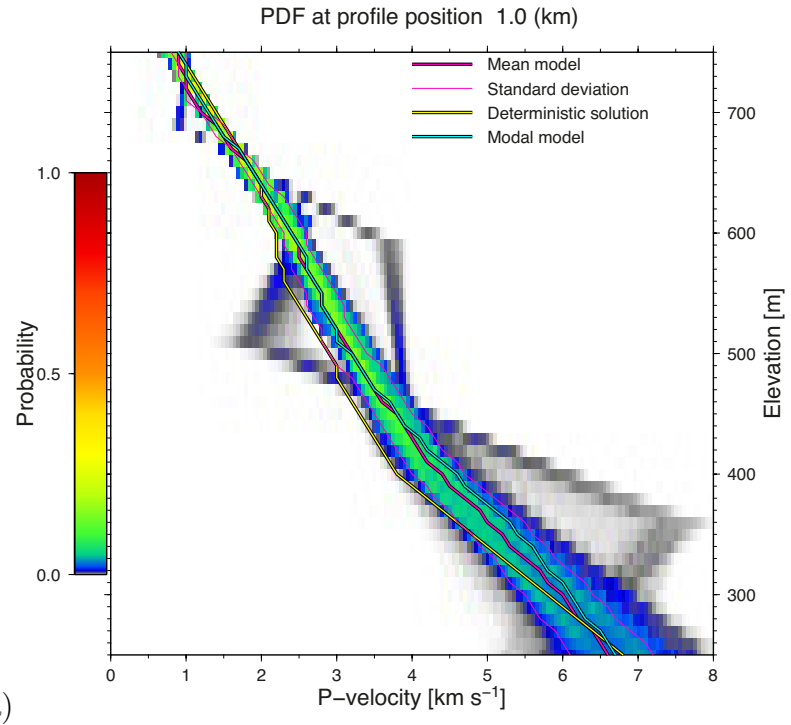


Figure 2.19: *Log-likelihood traces of the initial part for the 9 staggered chains. In the legend the sketch represents the shift direction of the staggered grids for each chain and the corresponding color of the traces.*

Differently from what was done in the inversion of our synthetic dataset we adopted a prior defined as uniform on the compact support $[0.3, 8.0]\, km/s$. This choice is based on the fact that the Salzach valley/infill geometry does not allow a more-informative depth-dependent prior since at the same elevation high velocities are

present (bedrock) together with lower ones (infill). The prior on the number of nodes was set as uniform in $[1, 59]$. The standard deviation of the proposal distribution (σ in eq. 2.6 and 2.7) is scaled at every node to a value that corresponds to 5% of the velocity obtained in the same position from the deterministic solution (2.18.b) used to initialize the sampling process. In this way the characteristic perturbation size is smaller for nodes that are expected to have lower velocity and bigger for nodes in faster areas of the model. Such a choice is meant to grant optimized mixing for all nodes, however other values for the proposal width σ would mainly influence the performances of the sampler, in terms of mixing properties and acceptance rate, thus the resulting posterior distributions are not supposed to be influenced by our choice. We discarded the first 1000 models of every chain (0.5% of all the accepted models), allowing the samplers to settle and oscillate around a stable number of inversion nodes, as well around a stable likelihood value. The fact that the chain related to the original grid (the grey trace in Fig.2.19) appears to sample better models (with lower misfits) should not turn out surprising: that parametrization is in fact the result of an optimization process aimed to identify an optimal parametrization, some of the grid shifts in the staggering process might indeed lead to less optimal parametrizations. This might result indeed in sampled models characterized by an average lower level of data fit. It might in other cases lead as well to the opposite situation.

Figure 2.20: *continues on next page*



(a)

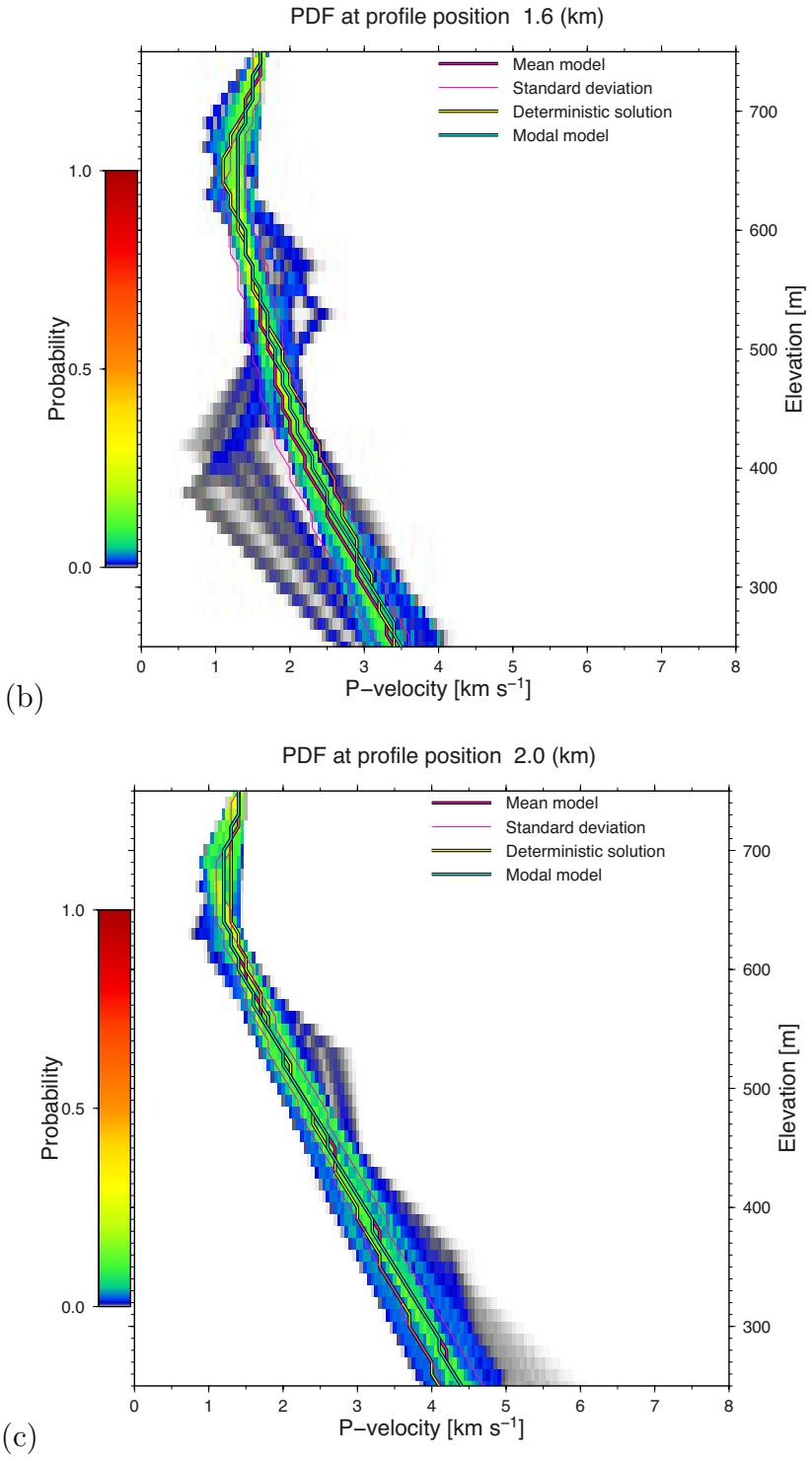


Figure 2.20: *PDF: vertical cross section at profile km 1.0 (a), 1.6 (b) and 2.0 (c) displaying the posterior velocity distribution as well as vertical profiles of the synthetic model, of the DLS solution, the mean posterior and its confidence interval delimited by one standard deviation.*

2.4.1.1 Posterior distributions

The *a posteriori* distributions of the sampled velocities are displayed as cross sections at profile coordinates 1.0 (Fig.2.20.a), 1.6 (Fig.2.20.b) and 2.0 km (Fig.2.20.c), together with the correspondent vertical velocity profiles extracted from the deterministic solution model, the modal and the mean solution with it's confidence interval defined with plus/minus one standard deviation. In order to ease the comparison between different methods, the FATT solution of Bleibinhaus and Hilberg (2012), computed on a static grid, is from here on substituted with a staggered FATT solution obtained with the same method, but applying in addition also the staggered grid approach described in the previous paragraph. This allows a fair comparison between models obtained from deterministic and probabilistic inversion methods. The PDFs at profile coordinates 1.0 (Fig.2.20.a) and 1.6 (Fig.2.20.b) show a multimodal behavior, with sets of low-likely models that deviate from the main, most-likely portion of the distribution. This effect is ascribed to the use of staggered grids: some of the staggered chains are spending some time sampling a less relevant part of the model space in the attempt to fit the data, given the staggered parametrization. Such an outcome is however not to be regarded as negative for the sampling process. The extremely low probability that characterizes this portion of sampled models leads them in fact to have a negligible influence on the inversion results.

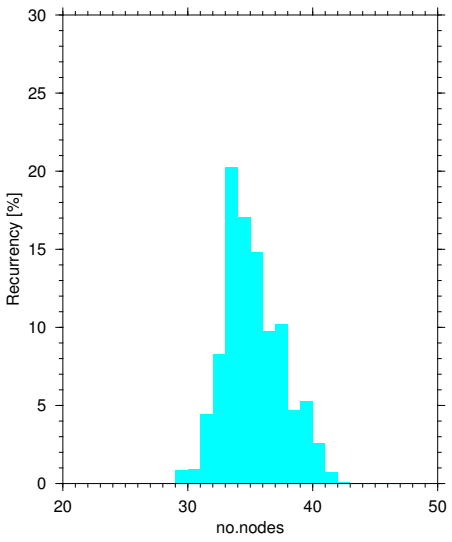


Figure 2.21: *Posterior distribution on the number of inverse parameters for the ensemble. The modal value corresponds to 33 nodes.*

The PDF of the number of nodes extracted from the models ensemble (Fig.2.21) points out that the sampling process clearly favored models parametrized with 33 nodes, a slight multimodal behavior is however to be noted with two minor peaks at 37 and 39.

2.4.1.2 Transdimensional inversion results

As done for our synthetic case-study, we extracted as a representative model solution the mean of the sampled ensemble. This *transdimensional McMC staggered mean model* (Fig.2.22.a) shows an overall good level of agreement with the solutions obtained through deterministic approaches: Full Waveform Inversion (Fig.2.22.b), Reflection-Refraction (Fig.2.22.c) and First Arrival Traveltime Tomography (Fig.2.22.d).
The shape of the Salzach valley bedrock we obtained yields some structural details that find correspondence in the FWI solution (green arrows in Fig.2.22.a and b) while they could not be recovered by the RRTT (whose parametrization of the reflecting interface intrinsically removes smaller details in the structure in favor of a regular structure) and FATT solutions. The interface between valley infill and bedrock, identified considering the 3.0 km/s isoline, shows a sharper velocity contrast in the McMC mean model compared to the FATT solution, where the same boundary appears more blurred.
The low velocity area (B) and an aquifer (A) are clearly outlined in our bayesian mean model and appear to generally confirm the solutions obtained with all the other methods. A higher level of similitude is however noticeable between the FWI and our solution, which are able to provide a level of complexity in the shape of the low velocity area not reached by RRTT and FATT. A minor zone with velocities lower than 1.5 km/s is present at the surface of the McMC solution between profile coordinates 2.2 and 2.5 km, corresponding to the right extremity of the area B as proposed by the FWI solution (blue arrow in Fig.2.22.a and b). Thickness and horizontal extension of the aquifer as recovered by our bayesian sampling process are compatible with the values proposed with all the other methods exception made for the FATT staggered solution that proposes slightly higher velocities in the near surface between km 2.0 and 2.5.
The 3.0 km/s isoline that could be considered as representing the boundary between valley infill and bedrock shows in the McMC solution a sub-elliptical feature that finds a mild correspondence in the FWI solution in a lower velocity area (red arrows in Fig.2.22.a and b).
The shape of the high velocity area highlighted with the letter (C) in the RRTT solution, seems to be qualitatively confirmed by all the other methods, however the estimated value of the velocity appears to be slightly lower for McMC, FWI and FATT model solutions, with values around 2.0 Km/s.

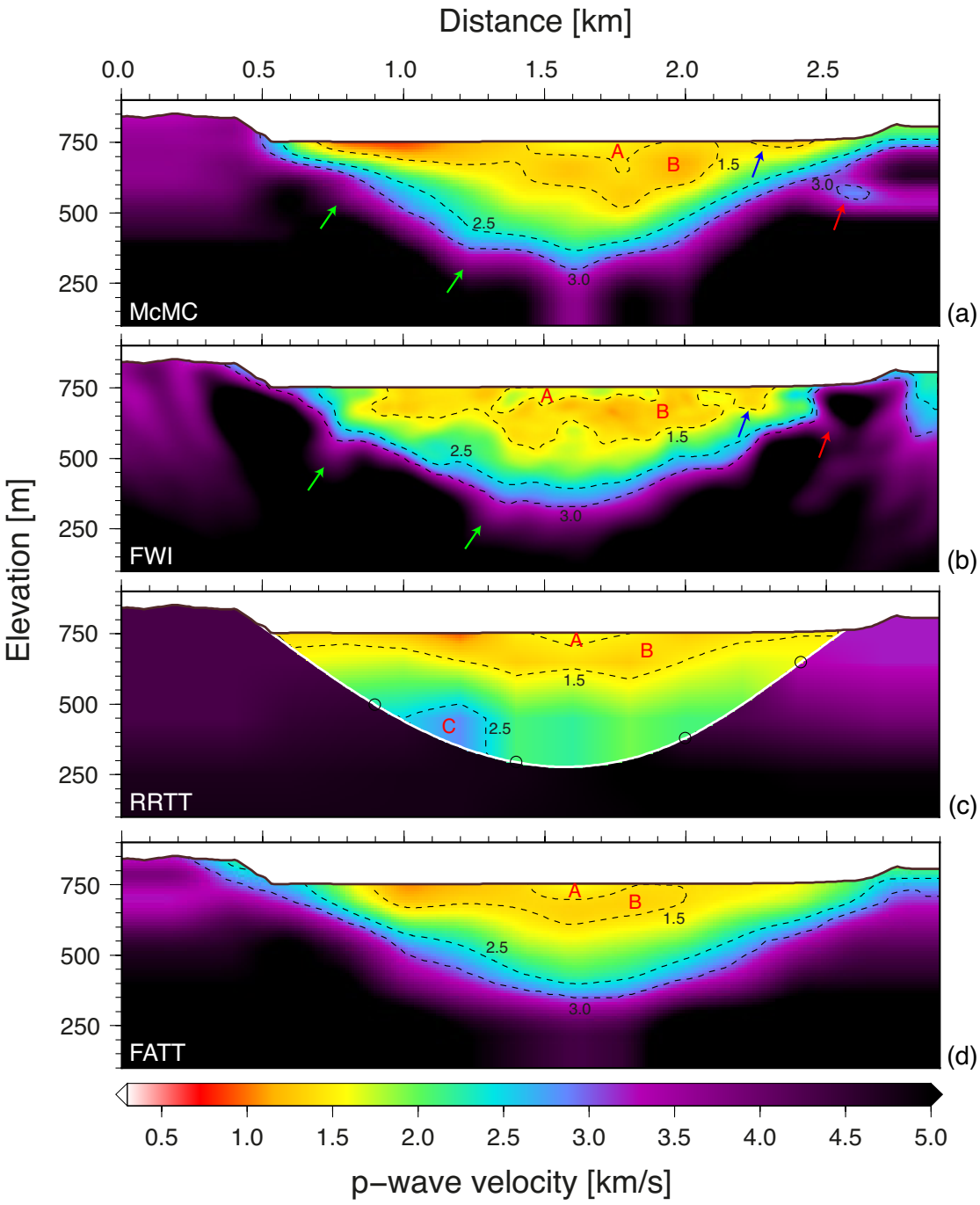


Figure 2.22: *Solution models compared: transdimensional McMC mean model (a), deterministic Full Waveform Inversion (b), deterministic Reflection-Refraction (c), deterministic first arrivals staggered (d). The arrows refer to features recovered with different methods, discussed in the text.*

2.4.1.3 Error maps

The quantitative uncertainty estimate of the solution made possible by the bayesian approach is expressed in Fig.2.23 in terms of a standard deviation map together with a relative error map.

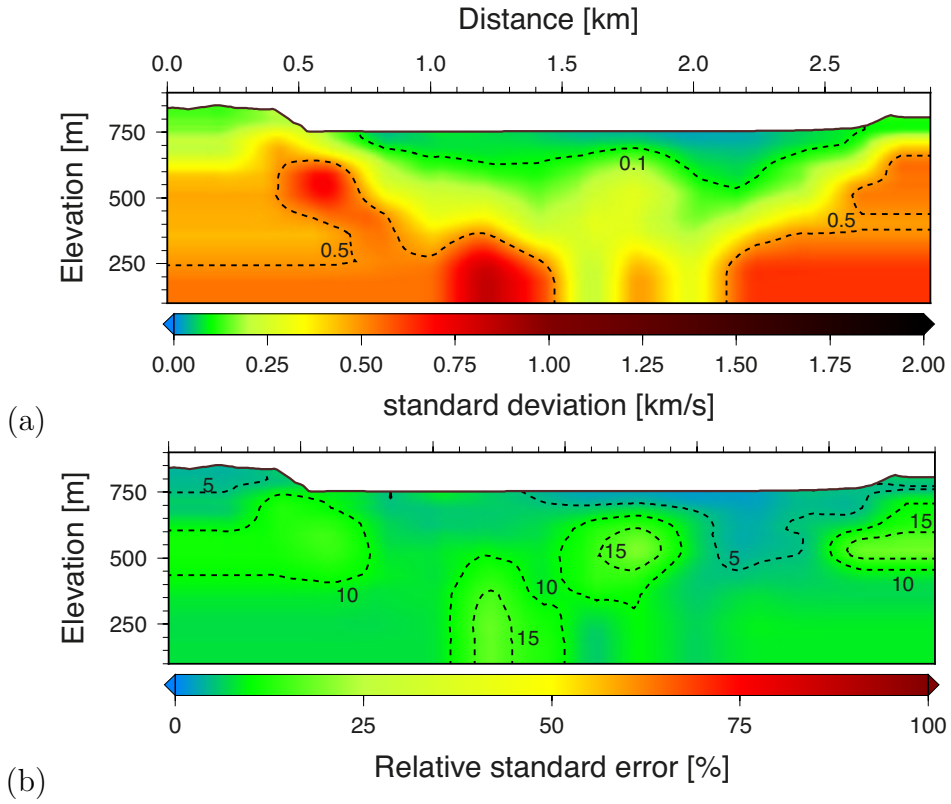


Figure 2.23: *Standard deviation (a) and relative error (b) maps.*

The uppermost part of the valley infill is characterized by uncertainties smaller than 0.1 km/s, while in depth the standard error doesn't exceed 0.5 km/s, value that could be considered as limit between infill and bedrock. The relative error map suggests that the horizontal anomaly located on the right side of the model at elevations between 500 and 600 m is likely to be an artifact. However, as we already pointed out in the previous paragraph, there is a good correspondence with the FWI solutions from Bleibinhaus and Hilberg (2012) where a discontinuity seems to characterize the bedrock at about 600 m and 2.5 km in profile direction. The area of higher uncertainty is located between km 1.0 and 1.4 on the deeper part of the model, here uncertainties up to almost 1 km/s are to be expected. The discontinuous values of estimated uncertainty in the bottom portion of the model (below 250 m.) raise a question on the validity of these estimates. At the same depth, in a portion of model identified as part of the bedrock, a relatively

wide range of uncertainties, spanning from ca. 0.20 to 1.0 km/s, appears odd. A further tool can help in this case to clarify the situation: a map displaying how often nodes located in a defined area have been treated as inverse parameters or were removed by the transdimensional sampling algorithm (Fig.2.24).

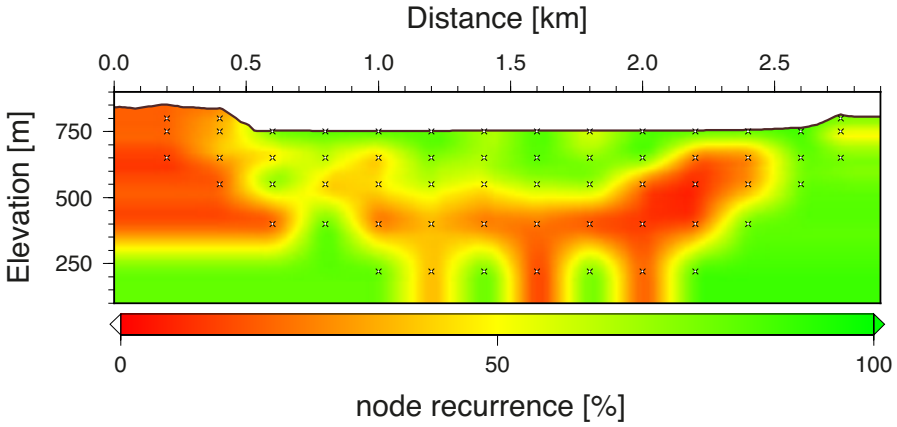


Figure 2.24: *Node recurrence map: the positions of possible inversion nodes (crosses) are displayed together with the relative frequency of each node being considered as an inverse parameter. Nodes that were more often set as inversion parameters have colors tending towards green, conversely less-often inverted nodes display colors towards red.*

Considering the bottom portion of the model, an alternating pattern is noticed, in correspondence to the aforementioned *discontinuous values of uncertainty*. Such a pattern points out that some node-configurations were more often preferred during the sampling process. This could be interpreted as an outcome of the *natural parsimony* property: a high parameter density is not needed at the bottom of the model and therefore models with a coarse node-distance are selected with higher frequency. Back to the issue of uncertainty estimation for the bottommost portion of the model: joining the information content of standard deviation (Fig.2.23.a) and node recurrency map (Fig.2.24) we suggest that an expected value for the standard deviation is to be computed as an average of the values characterizing the bottom nodes, corresponding to ≈ 0.5 km/s.
Proceeding with the uncertainty analysis, the low-velocity sub-elliptical structure marked with a red arrow in Fig.2.22.a) is likely to be an artifact, as suggested by the relative error map which proposes values of the relative error higher than 15%. The structure could correspond to a ripple in the infill/bedrock interface as proposed by the FWI solution.
Another area where the relative error reaches values of 15% involves the lower boundary of the low-velocity area (B) between km 1.7 and 1.8, with uncertainties in the order of 0.25 km/s expected to affect that portion of the model.

2.5 Conclusions

We presented an algorithm that allows for the inversion of first arrival traveltimes tomography data within a Bayesian framework. The process can be classified as a transdimensional Markov chain Monte Carlo, based on the Metropolis-Hastings algorithm. We adopted in addition a parallel-staggered grid approach that makes use of parallel-independent McMC processes and staggered grids in order to optimize the performances of the inversion.
The number and position of the inversion parameters is constantly changing during the sampling process thus is treated as an unknown of the inversion process. The amount and distribution of information in the data drives the choice of number and spatial location of the inverse nodes, relieving the operator from the need to chose and optimize a nor parametrization, neither smoothing or damping parameters. The solution models obtained appear naturally smooth as a result of the spatial averaging process over the sampled ensemble.
The practice of running parallel-independent sampling processes is proved to be very effective in the framework offered by the `simulr16` code. The extremely short burn-in phases we observed leads in fact to a speedup that yields an almost linear dependence with the number of parallel processes involved in the sampling. Making use of multiple instances of our McMC code reduces thus the run time needed to collect a satisfying number of models of a factor proportional to the cores/CPUs involved.
A staggered grid approach was adopted, as a compensation for the relatively coarse spatial distribution of the inversion nodes that characterizes the `simulr16` code. Staggered coarse grids allow to restrain the cardinality of the model space, thus result in shorter runtime needed to reach convergence compared to the corresponding single fine-grid alternative. Furthermore models obtained through a staggered-grid approach proved to reconstruct finer structural details without increasing the uncertainty of the solution.
Our code was first tested on a synthetic dataset, and then utilized to invert field data from the Salzach Valley (Austria) comparing our McMC solution with some results obtained by Bleibinhaus and Hilberg (2012) from the same data. The results obtained show a general agreement with the solutions proposed in the paper of Bleibinhaus and a particularly good match is observed with the Full Wave Inversion model even if our method makes use of first arrivals data only.
The inversion with our transdimensional McMC code offers of possibility of a quantitative uncertainty analysis by means of standard deviation and relative error maps. A useful tool for the interpretation of the uncertainty maps is provided by the node recurrence map.

Chapter 3

Resolution Matrix for a Multivariate updating scheme

3.1 Introduction

One of the most important issues of interpreting seismic tomography models is the need to provide a quantification of their uncertainty. Bayesian approach to inverse problems offers a rigorous way to quantitatively estimate this uncertainty at the price of a higher computation time. Optimizing bayesian algorithms is therefore a key problem. In this chapter we present a multivariate model-updating scheme which makes use of the constraints provided by the Model Resolution Matrix, aiming to a more efficient sampling of the model space. The Resolution Matrix relates the true model to the estimate, its off-diagonal values provide a set of trade-off relations between model parameters used in our algorithm to obtain optimized model updates. The bayesian algorithm we implemented and described in the previous chapter belongs to an important class of Markov Chain Monte Carlo methods called Random Walk Markov Chains. The trajectory of "steps" between neighboring states is a random walk and can be described with a graph in the model space. Such sampling schemes based on the Metropolis-Hastings algorithm require an accurate choice of proposal distributions: defining a good proposal distribution means choosing which size and kind of random steps is more convenient to efficiently move between neighboring model-states. As Metropolis et al. (1953) stated in one of the first papers that employed McMC sampling: "the maximum displacement must be chosen with some care; is too large most moves will be forbidden, and if too small, the configuration will not change enough. In either case it will take longer to come to equilibrium.". Anyway not only the perturbation size is key to an optimized algorithm, also the way one tries to move between models is of fundamental importance.

3.1.1 Optimization of Metropolis-Hastings McMC

In the first chapter of this work we focused on the formal basis that justify the importance of reversible, ergodic Markov chains. Ergodicity is, in a broad sense, a property of stochastic processes for which the time average over a sub-sequence of events corresponds to the global ensemble average. For Markov chains ergodicity guarantees the existence and unicity of a stationary distribution. Reversible Markov chains are satisfying the detailed-balance condition, which grants the existence of a stationary distribution. The behavior of reversible chains is independent from the time-direction of their evolution, they hold the same properties backwards of forwards in time.
The importance of these properties combined is that, for those who are seeking to implement, modify or, like us, optimize an updating scheme, it is fundamental to keep in mind that all the non-trivial known methods to construct updating mechanisms able to preserve a given equilibrium distribution, are special cases of the Metropolis-Hastings (M-H) algorithm that hold the reversibility condition (C.Geyer on Brooks et al. (2011)).
Furthermore since reversibility and ergodicity allow the applicability of the Markov chain central limit theorem (see eq.1.14) and asymptotic variance estimation, the preservation of such properties grants the possibility to estimate the expectations of some quantity related to the sampled ensembles (provided that the sampler is given enough time to reach convergence, i.e. to sample a sufficient number of models).
Optimal performances of McMC samplers based on the M-H algorithm are directly connected with the optimization of the M-H ratio (eq.1.27). The main points to consider are therefore:

- **Likelihood**: an appropriate function has to be utilized to quantify the ability of a model to fit observed data, this includes a proper estimate of the data noise. This work will not focus particularly on this aspect, however some interesting treatise and examples can be found in Pearse et al. (2009) and Bodin et al. (2012).

- **Prior**: a priori knowledge should be included in the algorithm with a probabilistic formulation, limiting where possible the model space to be sampled. Some consideration will be given in section 3.2.3.

- **Proposal**: how to efficiently and correctly propose new models. *Efficiency* is directly connected with the size of the "steps" between models while *correctness* relates to the updating scheme used, i.e. to how models are modified to produce new candidates.

In this chapter we will focus on this third point and propose a method that, through a multivariate updating mechanism, aims to increase the sampling efficiency while preserving fundamental properties that characterize Metropolis-Hastings updating schemes.

3.1.2 Comparing the efficiency of Markov Chains

Our purpose is, as said, to propose an optimized transition kernel (proposal distribution) that leads to a better McMC algorithm, therefore we first need to define what "better Markov chain" means and to point out some criteria that could be used to compare the efficiency of Markov Chains. Let's then lay down, following the lead of J. Rosenthal in Brooks et al. (2011, Chap.4), some rules to compare two chains $\{X_i\}$ with transition kernels $P_1(x, A)$ and $P_2(x, A)$ both with the same stationary distribution π: convergence speed, variance and mixing properties.

Convergence speed: Using the measure of *variational distance* introduced with eq.(1.8), the chain with kernel P_1 shows a faster convergence than P_2 if:

$$sup_A \left|P_1^n(x, A) - \pi(A)\right| \leq sup_A \left|P_2^n(x, A) - \pi(A)\right| \qquad \forall n, x \tag{3.1}$$

In other words, if at the n^{th} iterate of the transition kernel (eq.1.4), the conditional distribution of the chain P_1 is closer to the equilibrium distribution than the chain P_2, this means that the chain P_1 has been converging faster than P_2.

Variance: As a second criterion we consider some appropriate functional $g(X_i)$ of the chain and use it to compute a variance in the form:

$$V(g) = Var\left(\frac{1}{n}\sum_{i=1}^{N} g(X_i)\right) \tag{3.2}$$

then the chain following the transition kernel P_1 is more efficient if it has a smaller variance than the chain following P_2. This definition could however be dependent on the choice of the functional g and on the initial distribution X_0. Usual practice is to compare chains in stationarity, therefore after a long runtime. Anyway for a sufficient number of iterates of a transition kernel, the *memory* of the initial distribution should be lost.

Mixing: The third and last criterion is probably the most intuitive since it basically states that a chain is better if it allows a faster sampling of the model space.

$$\mathbf{E}_{\|\Delta X\|} = \mathbf{E}\left[(X_n - X_{n-1})^2\right] \tag{3.3}$$

Where the function $\mathbf{E}_{\|\Delta X\|}$ measures the *distance* from one member of a chain X_n to the previous one X_{n-1}, this can be accomplished for example with any functional like $\frac{1}{N}\sum_{i=1}^{N}(X_i - X_{i-1})^2$ (N is the number of states in the chain considered for the measure). If the estimate of the expectation, is bigger for P_1 than for P_2 we will say that the first chain shows a *faster mixing*. The value of eq.(3.3) is computed on all the steps of the sampler thus rejected steps will give a null contribution and small accepted ones will not help that much either and will also push towards an overall slowly-sampling chain. These different definitions can be to some extent merged into the single relaxed statement that a *better* Markov chain samples the model space more efficiently if it reaches the equilibrium distribution faster, it doesn't extensively sample low-likelihood areas and it proceeds in the sampling with steps that at the same time are not so big to be often rejected nor so small to be irrelevant.
Further diagnostic tools we will utilize include a qualitative analysis of the trace plots of single model parameters. The time evolution of the sampled values for some selected nodes is plotted and analyzed to graphically highlight the mixing behavior of different transition kernels, i.e. how big and how frequent are the steps between parameter values. If with eq.(3.3) the whole model is taken into account, with a measure of distance that involves all the parameters, a more simplistic analysis can be performed observing single parameters. Basic principles and examples of trace plots analysis can be found in Brooks et al. (2011, pp.95-98).
Another aspect of analyzing mixing properties is to understand how the perturbations applied to a model reflect in the likelihood of a trial model. A proposed step between model-states could be significant in terms of *distance* (chosen an appropriate metric for eq.3.3), but could at the same time lead to a proposed state whose likelihood is close to that of the previous state and therefore more likely to be accepted. To understand this behavior we will take into account the probability distributions of likelihood differences (L_D) between current and trial models $L_D = L_C - L_T$ to visualize the size of the "likelihood-steps" together with their probability. Proposed model whose L_D is close to zero are more likely to be accepted, therefore comparing Markov chains, transition kernels leading to L_D-distributions more peaked around zero will be considered more efficient.

3.1.3 Model Resolution Matrix

Given a forward problem $\mathbf{d} = \mathbf{G} \cdot \mathbf{m}$ we can find, through a *deterministic* inversion, one generalized inverse solution, an estimated model:

$$\mathbf{m}^{est} = \mathbf{G}^{-g} \cdot \mathbf{d} \tag{3.4}$$

considering the true model $\mathbf{m}^{true}$ that once measured has produced the data set $\mathbf{d}$:

$$\mathbf{d} = \mathbf{G} \cdot \mathbf{m}^{true} \tag{3.5}$$

and substituting eq.(3.5) into eq.(3.4) we obtain

$$\mathbf{m}^{est} = \left[\mathbf{G}^{-g}\mathbf{G}\right] \mathbf{m}^{true} \tag{3.6}$$

The Model Resolution Matrix can then finally be defined as

$$\mathbf{R} = \mathbf{G}^{-g}\mathbf{G} \tag{3.7}$$

Let us now point out that $\mathbf{R}$ is a function only of the data kernel $\mathbf{G}$ and of the a priori information added to the inverse problem to obtain $\mathbf{G}^{-g}$. Like $\mathbf{G}$, also $\mathbf{R}$ is a function of the parametrization that has been chosen to discretize a continuous system (Menke, 2012, pp.72-73). The Model Resolution Matrix (we will sometime make use of the abbreviation *ResM*) relates the true model with its estimate, and can be regarded as a filter through which the "real world" is observed.

$$\mathbf{m}^{est} = \mathbf{R} \cdot \mathbf{m}^{true} \tag{3.8}$$

Off-diagonal entries indicate the amount and direction of correlation between model parameters. Significantly far-from-zero off-diagonal entries point out strong trade-offs in the variables, whereas diagonal entries give information on the resolution of the model parameters *per se*, how much a parameter is constrained by the data.
Let's now give an explicit form to $\mathbf{R}$ for a Damped Least Squares (DLS) inversion strategy, that we will be using in this study. Using eq.(3.7) and Levenberg-Marquardt solution, which provides a generalized inverse in the form of eq.(1.18), we obtain:

$$\mathbf{R} \overset{\text{DLS}}{=} \left[\mathbf{G}^T\mathbf{G} + \theta^2\mathbf{I}\right]^{-1} \mathbf{G}^T\mathbf{G} \tag{3.9}$$

$\mathbf{R}$ is then a square $M \times M$ symmetric matrix, where M is the number of parameters used to discretize the measured system. Without damping the resolution matrix would be the unit matrix and our model estimate would be called *unbiased*. Seismic tomography unfortunately requires damping, the information carried by a seismic

ray is an integral over the whole ray path, influenced by several parameters. The better a parameter is resolved, the more $\mathbf{R}$ will tend to the unity matrix. Explicitly writing the i^{th} row of eq.(3.8) we have:

$$m_i^{est} = \sum_{j=1}^{M} R_{ij} \cdot m_j^{true} \tag{3.10}$$

where we can see that the estimate of each model parameter is a weighted average of the whole true model. The i^{th} row of $\mathbf{R}$ provides the weighting factors for the i^{th} parameter. The trace of $\mathbf{R}$ is sometimes regarded as an estimation of the degrees of freedom in the model, which is considered as the number of parameters that the dataset can resolve. For an ideally *unbiased* model, when $\mathbf{R} = \mathbb{1}$, the number of degrees of freedom would equal the number of model parameters M, but for almost every geophysical underdetermined problem the off diagonal elements of the resolution matrix will be non-zero and $\sum_{j=1}^{M} R_{ij} < 1$ (Nolet, 2008, p.278). Such situation, as for the example pictured in Fig.3.1, indicates the intrinsic limits of the inversion process and the inability to separate the effects and influence on the i^{th} parameter of all the others.

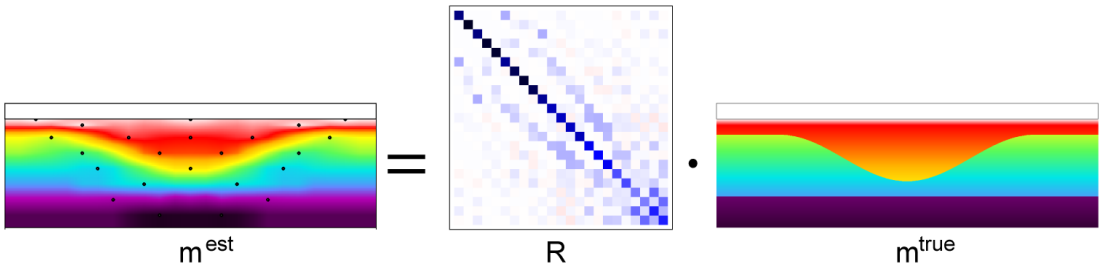


Figure 3.1: *A graphical representation of equation 3.8 relating a synthetic seismic velocity model (right) to one possible DLS-solution (left) through the model resolution matrix (middle). The i^{th} row of $\mathbf{R}$ shows how a perturbation of the true synthetic model will be mapped into the inverse parameters of $\mathbf{m}^{est}$. Well resolved inverse parameters have higher diagonal-element values ($R_{ii} \approx 1$, darker colors), poorly resolved parameters with almost zero resolution will tend to white.*

In this study we will make use of the information carried by the resolution matrix $\mathbf{R}$. This information reflects the uncertainty connected to the nature of the forward problem itself: how density and distribution of the data are expected to be resolved given a certain parametrization. Trade-off relations between model parameters will be exploited in the next section in order to implement a more efficient McMC updating scheme for a Markov chain Monte Carlo algorithm.

3.2 Method

So far we outlined the purpose of this research project and some of its theoretical basis. Let us now come to the implementation. As for the transdimensional algorithm in Chapter 2, we chose `simulr16` as the environment where the Bayesian inversion code has been embedded. This allowed us to exploit well established routines and parts of the inversion algorithm as well as to start off our Bayesian algorithm with a standard damped-least-squares (DLS) deterministic inversion. In the following sections four McMC with different transition kernels will be compared to a classical M-H McMC; operational criteria used for for the evaluation and determination of an optimal strategy will be the acceptance rate, the analysis of likelihood-differences distributions, the analysis of the mixing properties and of the posterior distributions.

3.2.1 Test model

The synthetic model features a 3-layered structure with smooth interfaces. The p-waves velocity in each layer is characterized by a 0.5 s^{-1} vertical velocity gradient. The uppermost layer shows a symmetrical synclinal structure.
The algorithm and results in this study are tested on the simple synthetic model pictured in Fig.3.2 together with the ray distribution for three sources. The total length of the model is 120m. and its maximum depth is 36m. The acquisition geometry consists of 23 sources and 23 receivers evenly distributed on the surface with a 5-meter spacing. The synthetic travel times, as done for the test-model in the previous chapter, have been computed using the FAST algorithm. Gaussian random noise has been added to the data using a standard deviation of 5% of the noiseless travel time.

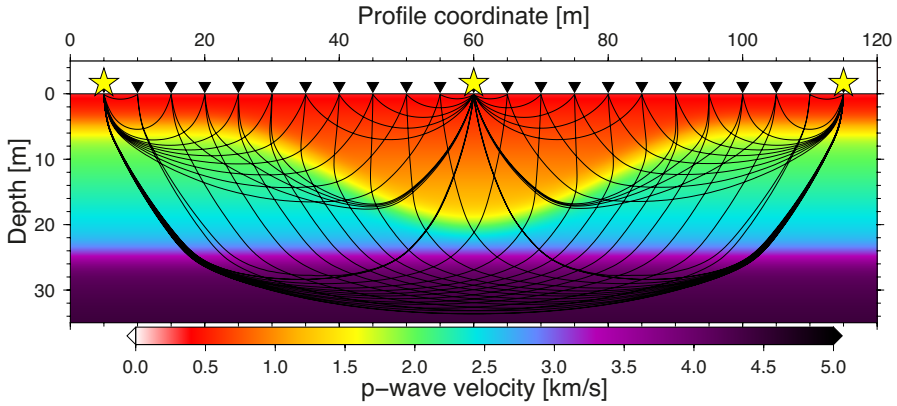


Figure 3.2: *Synthetic model with ray paths relative to the sources 1, 12, 23 (star). The receivers positions are marked with a triangle.*

3.2.2 Deterministic inversion

A damping test is performed and the parametrization adapted choosing an optimal node distribution with the same general workflow that has been described in Bleibinhaus and Hilberg (2012). The solution obtained is then used as the starting model in our McMC. This particular choice of the deterministic solution as initial state for the Markov chain as well as the decision to use an adapted parametrization are of course arbitrary choices, but we can nonetheless motivate them with some considerations.

First of all on the use of the deterministic solution as starting model for the Markov chain: any valid starting point we could choose is as good as any other point in the model space that might have been randomly picked out of the prior distribution. Any velocity model is allowed to be chosen as the first member of a Markov chain, so why not starting from a point of the model space that has a high probability of being close, if not inside, the equilibrium distribution Brooks et al. (2011). Since of course such a distribution is unknown before the sampling process we can assume that a solution of a DLS inversion will likely have the desired property of *vicinity* to the equilibrium distribution. This will have the positive effect of shortening the burn-in phase of the Markov chain at the negligible price of a deterministic inversion: starting the chain in a state close to the equilibrium saves us the time that would have otherwise been spent by the sampler accepting many low-likelihood models in order to approach the tail of the stability distribution. A further reason for us to commence the sampling from the DLS solution will be made clear in the following sections and it will be shown to be a fundamental requirement for our inversion process. A second point that deserves consideration is the use of an adapted parametrization: the spatial distribution and number of model parameters used in the DLS inversion leading to the initial model is optimized, and corresponds to the inverse parametrization used for the Bayesian inversion. As the aim of the study exposed in this chapter is to probe the feasibility, the effects, and eventually the benefits of a Resolution-Matrix-based transition kernel, we preferred not to employ a transdimensional approach, in order to simplify the problem and to better isolate the influences of the ResM only. For this reason an optimized parametrization offers some benefits. A lower number of parameters is needed, with a consequent minor computational load and faster execution of tests.

In addition to the reasons provided in the beginning of this section, the synthetic dataset has been inverted also in order to compare the results of our algorithm to a classical damped least squares inversion. The inversion process is started with a dense rectilinear node grid, the least resolved nodes are removed from the parametrization and the process (inversion-adaptation) is repeated until a satisfying compromise is reached between grid density and resolution.

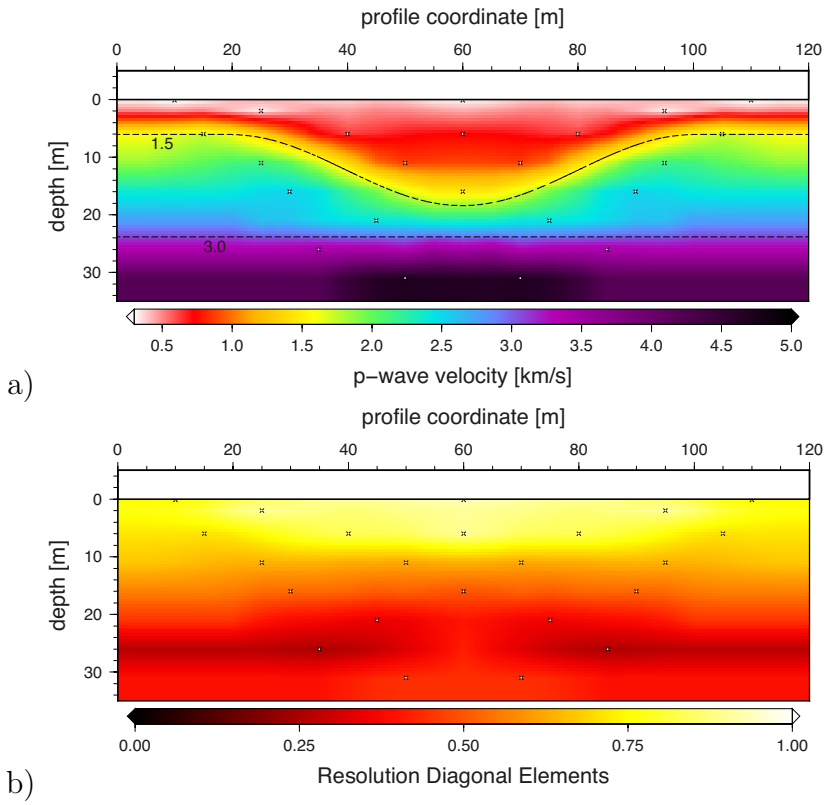


Figure 3.3: *a) Deterministic DLS solution: the dashed-black lines are contour lines of the synthetic model for the velocity values 1.5 and 3 Km/s. b) RDE map of the model solution above: the Resolution Diagonal Elements were used to optimize the distribution of nodes (black crosses)*

The linearized inverse solution (Fig.3.3.a) appears to reconstruct the layered structure of the synthetic model and the synclinal shape of the upper layer. The velocity field shows some minor discrepancies: the interface between first and second layer has been over smoothed, and the near surface nodes give underestimate the actual velocity. The resolution, as expected, exhibits higher RDE values in the upper part of the model.

3.2.3 Prior

With *prior*, or *a priori* information $p(\mathbf{m})$ we refer to any kind of information on the model $\mathbf{m}$ that we can include in our inversion process that is independent from measurements (Tarantola, 2005). In Bayesian formulation of seismic inverse problems, the prior information is all the knowledge we have of the velocity field, that doesn't come from the data we are going to process. Source of a priori informa-

tion could be previous studies, physical knowledge, or simply a reasonable range to which we want our velocity field to be limited. In this work a low-informative prior (Gelman, 2006) was used to limit the p-waves velocity to a reasonable depth-dependent range. Operatively prior-velocity intervals were defined both on the surface and on the bottom of our model, to obtain the prior at each depth through linear interpolation as in Fig.3.4.

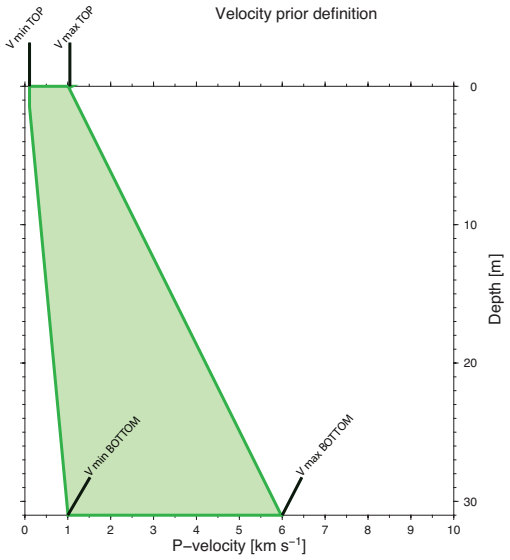


Figure 3.4: *Prior information: velocity ranges are defined both on the surface and on the bottom of the model to obtain the prior at each depth through linear interpolation.*

The prior probability distributions $p(v_i)$ for the model parameters are only dependent on the defined velocity ranges at each depth:

$$\Delta v_p(z) = v(z)_{max} - v(z)_{min} \tag{3.11}$$

thus we can define the prior:

$$p(\mathbf{m}) = p(\mathbf{v}) = \prod_{i=1}^{N} p(v_i) \tag{3.12}$$

$$p(v_i) = \begin{cases} 1/\Delta v_p(z), & \text{if} \quad v_i(z) \in \Delta v_p(z) \\ 0, & \text{otherwise} \end{cases} \tag{3.13}$$

3.2.4 Proposal and updating scheme

This is the hearth of a M-H-McMC: how to efficiently generate new models along a chain. As we introduced in the previous sections we will make use of the Model Resolution Matrix in order to obtain a more efficient updating scheme. Let $\mathbf{m}$ be the current model, a proposal is made for the new trial model $\mathbf{m}'$ drawing it

as a random deviate using a probability density $q(\mathbf{m}'|\mathbf{m})$ and then computing a *compensation term*. In classical M-H perturbation schemes as in eq.(3.14) only one component (the i^{th}) is updated at a time using a Gaussian probability density (3.15) :

$$\mathbf{m}' = \mathbf{m} + u\sigma_i \mathbf{e_i} \tag{3.14}$$

$$q(\mathbf{m}'|\mathbf{m}) \propto k \cdot exp\left(-\frac{1}{2}(\mathbf{m}-\mathbf{m}')^T \Sigma^{-1}(\mathbf{m}-\mathbf{m}')\right) \tag{3.15}$$

Here u is a normally-distributed random deviate from $N(0,1)$ and σ_i is an element of the matrix $\Sigma = diag(\sigma_1^2, \ldots, \sigma_n^2)$, that represents a standard deviation of the proposal (whose choice will be debated in the following section). Since in this chapter we don't consider transdimensional McMC, the parametrization of the models is not a variable in our inversion process, thus $\mathbf{m} = \mathbf{v} = (v_1, \ldots, v_n)$. The last two equations can then be re-written for the i^{th} component only:

$$v_i' = v_i + u\sigma_i \tag{3.16}$$

$$q(v_i'|v_i) = \frac{1}{\sigma_i\sqrt{2\pi}} exp\left\{-\frac{(v_i'-v_i)^2}{2\sigma_i^2}\right\} \tag{3.17}$$

In our algorithm the trial model will be instead proposed as:

$$\mathbf{m}' = \mathbf{m} + u\sigma_i \mathbf{e}_i - u\sigma_i \sum_{j\neq i}^{n} g(R_{ij})\mathbf{e}_j \tag{3.18}$$

where $g(R_{ij})$ is a parameter that is computed using some functional of the Model Resolution Matrix. Different functionals have been tested and will be exhaustively described and discussed in section 3.3 . What should be immediately clear is that our updating scheme modifies the velocity values of every model parameter. We can therefore regard eq.(3.18) as a *multivariate updating scheme*, that can be logically decomposed into a *perturbation term* which has the same form of eq.(3.14) and some resolution-matrix-based *compensations* whose effects are graphically illustrated in Fig.3.5.
The proposal probability density for updates in the form of eq.(3.18) has the same form as in eq.(3.15) since the *main perturbation* is drawn from a Gaussian distribution and the compensations have unit probability density, in other words given a perturbation to the i^{th} node, the compensations to the other nodes are defined from $g(R_{ij})$ only. Our resolution-matrix-based multivariate updating scheme retains therefore the great property of having a symmetric proposal ratio as will be proved in section 3.2.6.2.

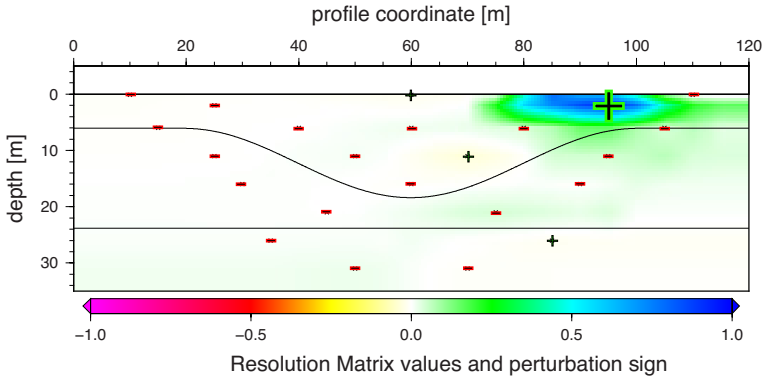


Figure 3.5: *Example: a perturbation to the 5th model parameter (big +) is balanced by opposite-sign compensations. The global biasing effect of a perturbation on the model is thus reduced, resulting in a model that is more likely to be accepted.*

3.2.5 Perturbation scaling

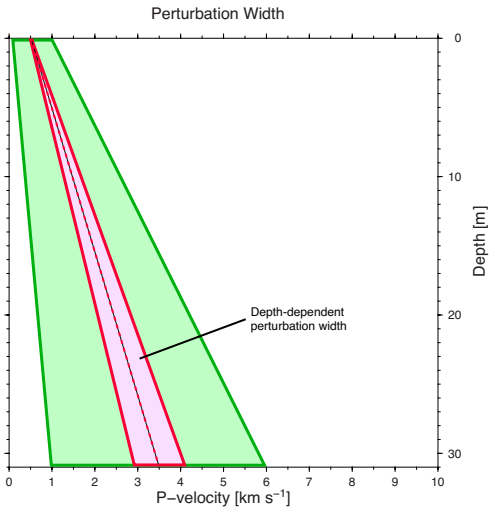


Figure 3.6: *Depth dependent perturbation size defined as a fraction of the prior range.*

Directly connected with the previous definition of velocity prior is the scaling of the perturbations size. The magnitude of a perturbation has a fundamental influence on the performances of Metropolis-Hasting based Markov chain Monte Carlo algorithms and needs to be carefully selected to ensure good mixing properties. The σ_i parameters in eq.(3.17), entries of the Σ matrix in eq.(3.15), are computed as a fraction of the prior range at each depth, in this way model parameters at the same depth will be perturbed with equally scaled proposal probability density:

$$\sigma_i = k \cdot \Delta v_p(z) \tag{3.19}$$

The value of the scaling constant k has been selected in order to achieve an acceptance ratio between $20 - 30\%$ in a classical M-H-McMC. That range has been taken as an optimum considering the study of Roberts et al. (1997) which proved the optimal acceptance rate to be exactly 23.4%, under some assumptions that will not be discussed in this work. Seven test McMC instances were run, characterized

z $[m]$	v_{min} $[Km/s]$	v_{max} $[Km/s]$	σ_i $[Km/s]$
0.1	0.10	1.00	0.090
2.0	0.16	1.31	0.115
6.0	0.27	1.95	0.168
11.0	0.42	2.76	0.235
16.0	0.56	3.57	0.301
21.0	0.71	4.38	0.367
26.0	0.85	5.19	0.434
31.0	1.00	6.00	0.500

Figure 3.7: *Table reporting the node depths and the corresponding prior velocity intervals together with the standard deviation of the random perturbations.*

respectively from the scaling constants in the set $\{0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1.0\}$. The acceptance rate has been monitored and related to k as in Fig.3.8 to determine its optimal value.

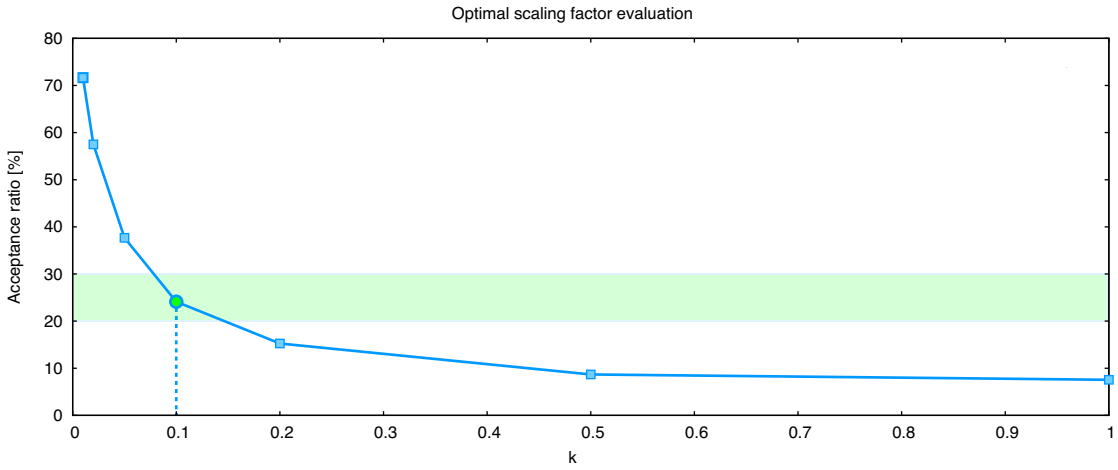


Figure 3.8: *Acceptance rate / k-factor relations: the selected value for the perturbation scaling factor is $k = 0.1$ and corresponds to $1/10$ of the velocity prior. The green area marks the optimal range characterized by a $20 - 30\%$ acceptance rate.*

3.2.6 Algorithm implementation

In section 3.2.4 we defined a way to perturb the current model $\mathbf{m}$ to propose a new candidate model $\mathbf{m}'$ for our Markov chain. Let us now describe schematically how equation 3.18 was implemented into a Bayesian-inversion algorithm. We developed and tested two candidate algorithms that will be referred to as Full-ResM and Fix-ResM McMC respectively. The latter proved to work correctly and has been

utilized in the rest of this study while the first showed a number of flaws that led to the decision not to carry on with its development.

3.2.6.1 Full-ResM updating scheme

1. Perform a DLS inversion to obtain a generalized inverse solution $\mathbf{m}^{det}$ (deterministic solution Fig. 3.2.2)

2. Initialize the Markov Chain: the deterministic solution is set as starting model $\mathbf{m}^{det} = \mathbf{m}$.

3. Compute $\mathbf{R}$ the resolution matrix of the current model $\mathbf{m}$

4. Generate a new trial model $\mathbf{m}'$ perturbing the current model $\mathbf{m}$

   - At every iteration of the Markov process choose one of the n model parameters randomly from a uniform distribution $i \in u[1, n]$
   - Randomly perturb the velocity of the i^{th} model parameter drawing from a Gaussian proposal probability density $q(v_i'|v_i)$ (3.17)
   - Compute and apply compensations to the other $j \neq i$ parameters using $\mathbf{R}$ as in eq(3.18)

5. Solve the forward problem: compute the estimated travel times $\mathbf{d}_{obs}$ and evaluate the likelihood of the trial model $L_{trial} = p(\mathbf{d}_{obs}|\mathbf{m}')$

6. Metropolis step: with the classical M-H algorithm randomly decide wether to accept or reject the proposed move from $\mathbf{m}$ to $\mathbf{m}'$ with the accepting probability $\alpha(\mathbf{m}'|\mathbf{m})$ in eq.(1.26).

This schematized algorithm is the first intuitive way to modify a Metropolis-Hastings McMC including equation 3.18 in its updating scheme. A random perturbation is applied to the current state $\mathbf{m}$ of the chain and compensations are computed from a functional $g(\mathbf{R}(\mathbf{m}))$ that yields information on the trade-off relations between model parameters, given by the resolution matrix that therefore needs to be computed at every iteration of the Markov chain.
While proposing a new updating scheme for our modified M-H McMC we want to preserve the stationarity of the stochastic process, therefore the reversibility (i.e. detailed-balance condition) must be preserved as well. With a non linearly-modified perturbation rule as eq.3.18 the reversibility is no longer guaranteed, since R_{ij} is computed at every iteration, and the transition probability is dependent on the current state of the chain. We chose to ignore this in a first attempt, assuming the modifications of R_{ij} to be non-systematic, in the attempt to verify

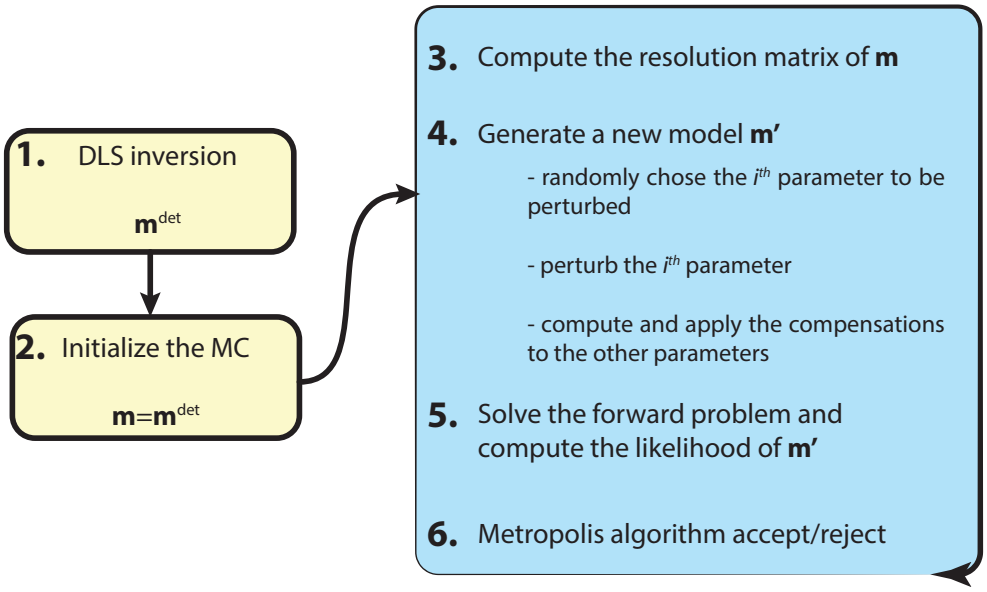


Figure 3.9: *Scheme of the Full-ResM McMC algorithm: this updating scheme includes the computation of the resolution matrix at every iteration of the Markov process.*

if the stationarity would be maintained. Under this assumption the proposal ratio becomes unity and the acceptance probability given by eq.1.26 would simplify to the likelihood ratio (eq.1.29).

We tried to justify this assumption with a simple test: we run a McMC setting the likelihood to a uniform distribution (Bodin and Sambridge, 2009), in this way we basically removed the data from the inverse problem and always accepted every proposed move. From Bayes theorem we know that in this case the posterior distribution sampled in the process should be directly proportional to the chosen prior distribution. Should this not be case we would have proved false our assumption.

The posterior distributions recovered, show no correspondence with the prior distribution that we selected, displayed here as the area delimited by the dashed red curves. In particular the PDFs in Fig.3.10.a and .b show a clear shift to lower velocities suggesting a higher probability of making moves to lower velocities, which contrasts with the expectation of uniform distributions. A comparison with the probability distributions obtained with the same test employing a classical M-H algorithm (without Resolution matrix) shows how we would expect PDFs from a "healthy" Markov chain to behave (Fig.3.10.b and .c). In this case the probability distributions appear to be simmetric respect to the middle of the prior area, and at depts where inverse nodes are located (red arrows) the distribution is uniform,

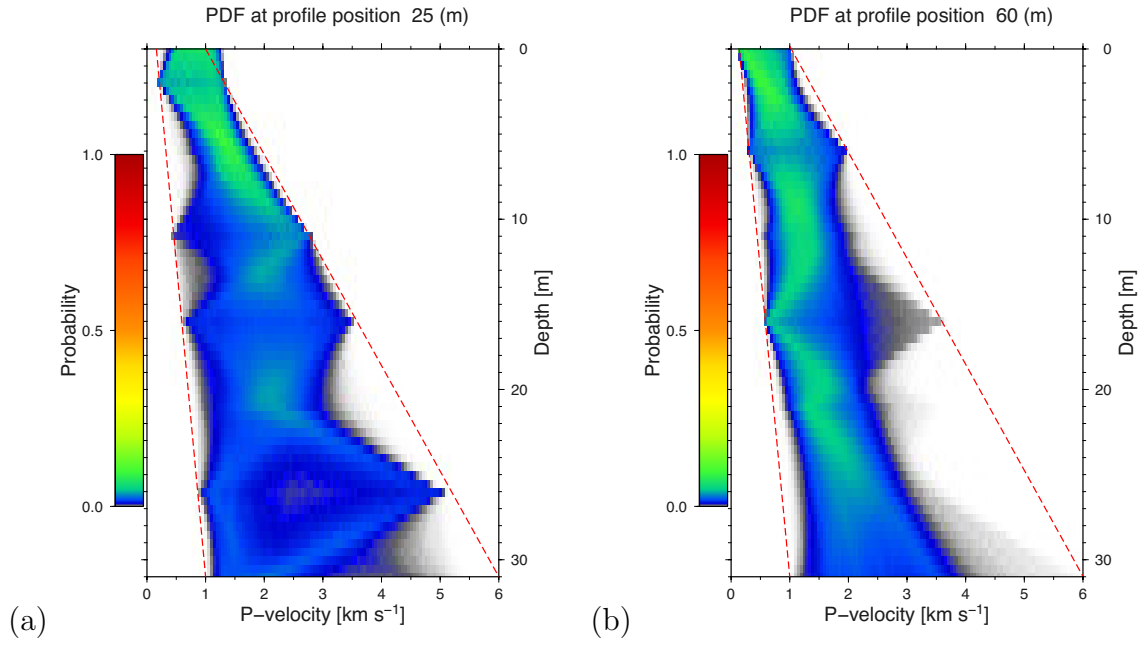
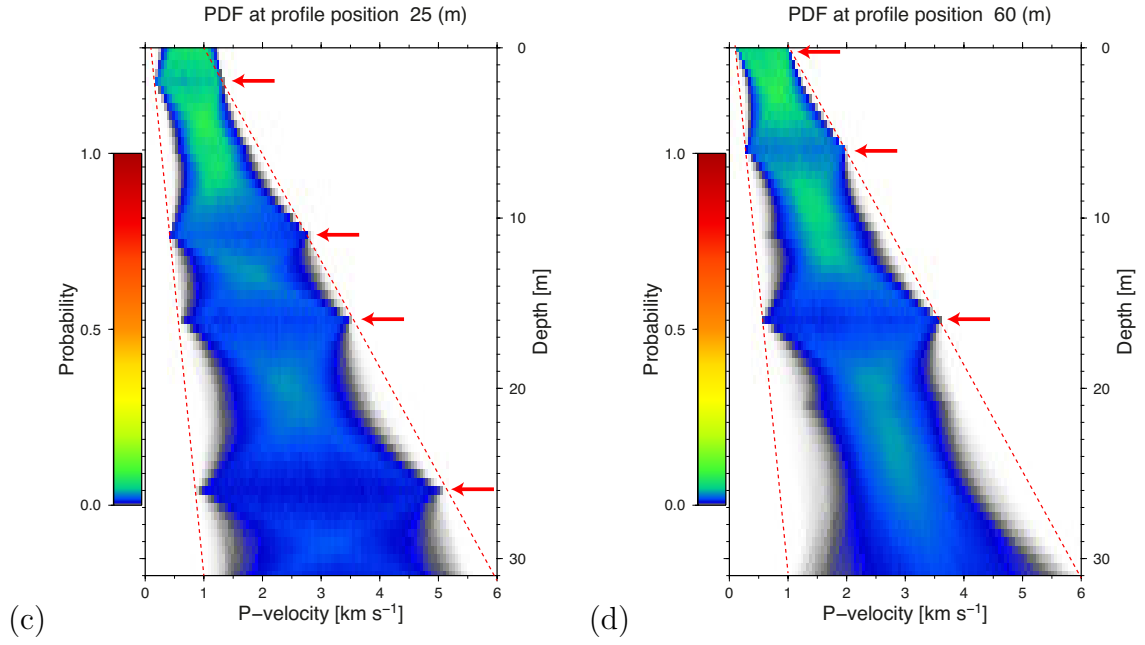


Figure 3.10: *Posterior distributions of the velocity displayed for profile position 25 (a) and 60 (b) obtained with the Full-ResM updating scheme. The area delimited by the dashed red curves corresponds to the prior.*



PDFs for profile position 25 (c) and 60 (d) obtained with a classical M-H algorithm. The red arrows indicate the dephts at which inverse nodes are located.

therefore the prior has been correctly sampled. Such an outcome can be considered as an heuristic proof of the fact that the transition kernel of our Full-ResM algorithm is not symmetric as we assumed, thus a McMC algorithm implemented as above would not be able to correctly sample the desired posterior distribution. Even without giving a complete formulation of the transition probability between states in a Markov chain characterized by the Full-ResM updating scheme, we can anyway state that it will be proportional to a term given by the functional $g(\mathbf{R})_n$:

$$q(\mathbf{m}_{n+1}|\mathbf{m}_n) \propto g(\mathbf{R})_n \tag{3.20}$$

where the index n signifies that $\mathbf{m}_n$ is the n^{th} state in the chain, $\mathbf{m}_{n+1}$ is the proposed state and $g(\mathbf{R})_n$ is the resolution-matrix functional of the n^{th} model/state. The dependency of the transition probability from a term that is adapted at every step of the Markov process leads to a state-dependent chain that does not converge to the desired posterior distribution. The state-dependency of the Markov chain isn't however the reason for the loss of convergence, since some drift-conditions could be enforced in order to grant Harris-recurrence (Tweedie and Meyn, 1993, pp.466). It is instead the continuous adaptation of $\mathbf{R}_n$, thus of the transition probability, that makes for a non-converging process. With such a continuous adaptation, the transition probabilities in a Markov chain can be expressed through a family of transition kernels as:

$$Pr(\mathbf{m}_{n+1}|\mathbf{m}_n, \mathbf{m}_{n-1}, \ldots, \mathbf{m}_0, K_n, K_{n-1}, \ldots, K_0) = P_k(m, A) \tag{3.21}$$

where K_n is the transition kernel at the n^{th} iteration, depending on $\mathbf{R}(\mathbf{m}_n)$. It then possible that the probability to move from $\mathbf{m}_n$ to $\mathbf{m}_{n+1}$ could depend from the whole history of the process and lead to the loss of convergence to a stationary distribution. Roberts and Rosenthal (2007) proved that asymptotic convergence can be however preserved under the *diminishing adaptation* condition:

$$\lim_{n\to\infty} \sup ||P_{K_{n+1}}(m, \cdot) - P_{K_n}(m, \cdot)|| = 0 \tag{3.22}$$

Here the requirement is that the adaptation at the n^{th} iteration goes to 0 as n tends to infinity. Adaptive McMC are well known and can be easily found in literature (e.g. Rosenthal in Brooks et al. (2011) chap.4) thus we will omit a more detailed theoretical digression. A first possible solution to the loss of ergodicity and convergence could be the introduction of a time dependent scaling factor in the transition probability to assure that its adaptation will diminish over time.

$$q(\mathbf{m}_{n+1}|\mathbf{m}_n) \propto g(\mathbf{R}_n) \cdot d(n) \tag{3.23}$$

In this way we would respect the condition (3.22) and save the convergence, but on the other hand we would lose at some point the trade-off relations carried by

the functional $g(\mathbf{R})_n$ used to compute the compensation terms in our multivariate updating scheme, which is the main purpose of this study.
Discarding the option of introducing a scaling factor, the next possible solution can be found dropping the practice of computing $g(\mathbf{R})_n$ at every iteration and evaluating it instead on batches of models of fixed length with a simple average (which still requires the computation of the resolution matrix at every iteration). In this way we would not be adapting the transition probabilities at every iteration, and within each batch the convergence to an equilibrium distribution would be granted by the central limit theorem. Now that with the introduction of batches we can avoid the issues connected with adaptation, we need to verify that a functional $g(\bar{\mathbf{R}}_b)$ could still be used to successfully compute the compensation terms in our updating scheme. In oder words, we know that the "exact" trade-offs are obtained only re-computing $\mathbf{R}$ at every step, but since that method has been proved unusable our goal is now to prove that an approximation holds enough information.
Provided that a batch has a "sufficient" length, the difference between the conditional distribution of each batch and the equilibrium distribution has to be small (zero for an infinite-length batch), hence we can assume that the $\mathbf{R}$ matrixes of the models in one batch are numerically not so dissimilar to each other and so is the average resolution matrix $\bar{\mathbf{R}}_{b,q}$ computed on that same batch. This assumption can be furthermore extended considering that an average on all the batches will also yield this similarity:

$$\bar{\mathbf{R}}_{b,q} = \frac{1}{b} \sum_{n=b(q-1)+1}^{bq} \mathbf{R}_n \tag{3.24}$$

where b is the number of models in a batch, Q is the number of batches and q is indexing them.

$$\bar{\mathbf{R}}_{b,1} \approx \bar{\mathbf{R}}_{b,2} \approx \ldots \approx \bar{\mathbf{R}}_{b,Q} \approx \frac{1}{q} \sum_{q=1}^{Q} \bar{\mathbf{R}}_{b,q} \tag{3.25}$$

The batch approach leaves us however with three main issues:

- the need to have "sufficiently" long batches to ensure stability
- the need to still evaluate $\mathbf{R}$ at every iteration
- the need to ensure that the diminishing adaptation condition (3.22) is respected

3.2.6.2 Fix-ResM updating scheme

The matrixes in eq.(3.25) are all Monte Carlo approximations of a functional of the resolution matrix, evaluations of a "quantity" that carries useful information on the

trade-off relations between model parameters. Computing such a quantity instead of evaluating it trough Monte Carlo approximation would be a solution to all the issues listed above that can be practically achieved by means of a deterministic linearized solution of the inverse problem assuming that eq.(3.25) can be extended as follows:

$$\frac{1}{q}\sum_{q=1}^{Q}\bar{\mathbf{R}}_{b,q} \approx \mathbf{R}(\mathbf{m}^{det}) \tag{3.26}$$

Adopting a simplistic language, equation (3.26) basically follows from the consideration that if we "trust" a linearized solution of an inverse problem to be an "acceptably good" solution, then we can also expect it to be similar to the true model and as well to a mean model $\bar{\mathbf{m}}$ extracted from the Markov chain (see Section 1.7.5). What if the above statement happens to be false and the deterministic solution doesn't show such a congruence? In that case the updating scheme efficiency would not benefit from correct trade off-relations between the inverse parameters, nonetheless the sampling process would be carried on properly and would eventually sample the target distribution even if with a lower efficiency.
Our *fixed resolution matrix* updating scheme (Fix-ResM McMC) has been therefore implemented as follows:

1. Perform a DLS inversion to obtain a generalized inverse solution $\mathbf{m}^{det}$ (deterministic solution Fig.3.3)

2. Compute $\mathbf{R}^{det}$ the resolution matrix of $\mathbf{m}^{det}$

3. Initialize the Markov Chain: the deterministic solution is set as starting model $\mathbf{m}^{det} = \mathbf{m}$.

4. At every iteration of the Markov process: generate a new trial model $\mathbf{m}'$ perturbing the current model $\mathbf{m}$

   - Choose one of the n model parameters randomly from a uniform distribution $i \in u[1, n]$
   - Randomly perturb the velocity of the i^{th} model parameter drawing from a Gaussian proposal probability density $q(v_i'|v_i)$ as in eq.(3.17)
   - Compute and apply compensations to the other $j \neq i$ parameters using $\mathbf{R}^{det}$ as in eq.(3.18).

5. Solve the forward problem: compute the estimated travel times $\mathbf{d}_{obs}$ and evaluate the likelihood of the trial model $L_{trial} = p(\mathbf{d}_{obs}|\mathbf{m}')$

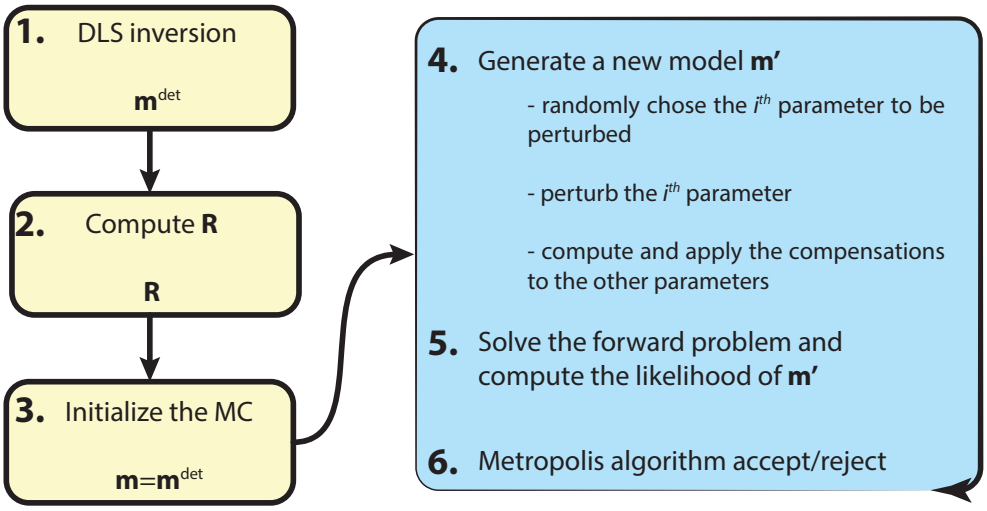


Figure 3.11: *Scheme of the Fix-ResM McMC algorithm: the steps grouped in the light blue box constitute the iterative McMC process, whereas the first three points are part of the deterministic initialization of the inversion.*

6. Metropolis step: with the classical M-H algorithm randomly decide wether to accept or reject the proposed move from $\mathbf{m}$ to $\mathbf{m}'$ with the accepting probability $\alpha(\mathbf{m}'|\mathbf{m})$ given in eq.(1.29).

With the above scheme there is no need to compute the resolution matrix $\mathbf{R}(\mathbf{m}^k)$ from each k^{th} state of the chain as this results in an increased computational load and yields yet all the issues we dealt with in the previous section. The increased computational load can be easily ascribed to the necessity to compute the ray tracing at every McMC iteration, since to derive $\mathbf{R}$ we need to update the values of $\mathbf{G}$ for every $\mathbf{m}^k$ in eq.(3.9). A comparison of the CPU-cost of the two methods applied on our synthetic model reveals that in average every McMC iteration of the Fix-ResM algorithm takes approximatively 1.225 seconds, whereas the Full-ResM needs 1.445 seconds. The latter proves then to be an almost 20% slower algorithm.

3.3 Tests and results

The Fix-ResM compensation scheme has been implemented, tests and results will be presented and discussed in this section. So far we never explicitly gave an expression of the compensation terms in eq.(3.18), using instead the functional $g(\mathbf{R})$. Four candidates were proposed [1]:

$$\mathbf{a}) \qquad g(\mathbf{R}) = \sum_{i \neq j}^{n} R_{ij} R_{ii} \tag{3.27a}$$

$$\mathbf{b}) \qquad g(\mathbf{R}) = \sum_{i \neq j}^{n} \frac{R_{ij}}{R_{ii}} \tag{3.27b}$$

$$\mathbf{c}) \qquad g(\mathbf{R}) = \sum_{i \neq j}^{n} R_{ij} \tag{3.27c}$$

$$\mathbf{d}) \qquad g(\mathbf{R}) = \sum_{i \neq j}^{n} \frac{R_{ij}}{\sum_{j \neq i}^{n} R_{ij}} \tag{3.27d}$$

which substituted in eq.(3.18) give the explicit equations for perturbation and compensations to generate a proposed model

$$\mathbf{a}) \qquad \mathbf{m}' = \mathbf{m} + u\sigma_i \mathbf{e}_i - u\sigma_i \sum_{i \neq j}^{n} R_{ij} R_{ii} \mathbf{e}_j \tag{3.28a}$$

$$\mathbf{b}) \qquad \mathbf{m}' = \mathbf{m} + u\sigma_i \mathbf{e}_i - u\sigma_i \sum_{i \neq j}^{n} \frac{R_{ij}}{R_{ii}} \mathbf{e}_j \tag{3.28b}$$

$$\mathbf{c}) \qquad \mathbf{m}' = \mathbf{m} + u\sigma_i \mathbf{e}_i - u\sigma_i \sum_{i \neq j}^{n} R_{ij} \mathbf{e}_j \tag{3.28c}$$

$$\mathbf{d}) \qquad \mathbf{m}' = \mathbf{m} + u\sigma_i \mathbf{e}_i - u\sigma_i \sum_{i \neq j}^{n} \frac{R_{ij}}{\sum_{j \neq i}^{n} R_{ij}} \mathbf{e}_j \tag{3.28d}$$

In the next sections the four proposed transition kernels associated to the above equations have been tested in order to establish which resolution matrix functional gives the optimal perturbation scheme. The letter notation "functional-a/b/c/d" will be used from now on with reference to the above four candidates expressed in

[1] the choice of these specific four candidates is to some extent heuristic: we will argue in the conclusions that more functionals could be proposed and investigated.

equations (3.27a/3.27d) with the addition of the "non-ResM" functional indicating a classical perturbation scheme that makes no use of resolution-matrix based compensations.

3.3.1 Non-McMC tests

In order to compare the performances of the four functionals (3.27a/3.27d) while excluding the stochastic effects typically expected in a Monte Carlo process, a non-McMC test has been carried on as follows:

- The synthetic dataset is inverted with a McMC based on a *classical* Metropolis algorithm (no ResM-based perturbation scheme are used).
- 500 models are sub-sampled[2] from the Markov chain ensemble obtaining a non-correlated subset.
- Each of the test models is perturbed iteratively using different methods: algorithm 2 displays a pseudo code.
- The models resulting from every perturbation with each method are mutually compared
- Statistical inference is performed in order to identify the optimal ResM-functional

Such a non-stochastic test finds its reason to be in the fact that a specific perturbation can be applied, employing every method under exam, while controlling its magnitude and excluding other non reproducible effects. In this way specific properties of each transition kernel can be examined. The magnitude of the perturbations is 10% of the prior range given in eq.(3.11) at the depth z_i of the perturbed node: $v_i' = v_i + 0.1 \cdot \Delta v_p(z_i)$. This choice has been made clear in section 3.2.5. Considering that every model parameter is perturbed twice (once with a positive velocity increment, once with a negative one), our non-McMC test produces five sets of 23000 perturbed models, one set for each transition kernel / functional being tested.

Misfit analysis The first of the properties compared to assess the performances of the five perturbation schemes is the ability to propose "better" models, namely to propose models whose misfit decreased after the perturbation proposed. Evaluating the percentage of models characterized by lower misfit in Fig.3.12 the

[2]Subsampling a Markov chain at spacing k is the process of taking every k^{th} element of a Markov chain $m_1, m_2, \ldots$ obtaining a new Markov chain $m_1, m_{k+1}, m_{2k+1}, \ldots$

Algorithm 2 Pseudo-code of the non-McMC test

for all models in the subset: $\mathbf{m} = 1, 500$ **do**
  for all parameters: $i = 1, n$ **do**
    perturb node i:
    $\Delta v_i = 0.1 \cdot \Delta v_p(z_i)$
    $v_i' = v_i + \Delta v_i$
    for all parameters $j \neq i$ **do**
      compensate node j:
      no-ResM) $v_j' = v_j$
      func-a) $v_j' = v_j - \Delta v_i \cdot R_{ij} R_{ii}$
      func-b) $v_j' = v_j - \Delta v_i \cdot {}^{R_{ij}}/_{R_{ii}}$
      func-c) $v_j' = v_j - \Delta v_i \cdot R_{ij}$
      func-d) $v_j' = v_j - \Delta v_i \cdot {}^{R_{ij}}/_{\sum_{j \neq i}^{n} R_{ij}}$
    end for
  end for
end for

functional-a (eq.3.27a) is identified ad the best performer with 13.30% of the proposed model showing a reduced misfit after being perturbed.
A more specific analysis was performed considering separately perturbations that involved three different groups of model parameters corresponding relatively to shallow, medium and deep nodes. A depth-based subdivision finds its justification in the values taken by the resolution diagonal elements of the deterministic solution:

- Shallow: RDE values are in the range 0.92 - 0.70 (nodes 1-10)
- Middle: RDE values are in the range 0.63 - 0.50 (nodes 11-17)
- Deep: RDE values are in the range 0.44 - 0.25 (nodes 18-23)

Deeper nodes in Fig.3.13 are less constrained, while moving towards the surface a higher ray density ensures the model parameters to be better resolved. Caveat: this very synthetic model has a relatively simple structure that reflects in a quite even ray distribution and depth-dependent resolution. In more complex cases relations between resolution, ray densities and depth would need different strategies to group nodes together. The sub-grouping allowed us to observe that the functional-c has slightly better performances on the middle nodes, where 17.71% of perturbations lead to lower-misfit models against the 16.43% of the functional-a that nevertheless showed the best overall performances. Above all the percentage

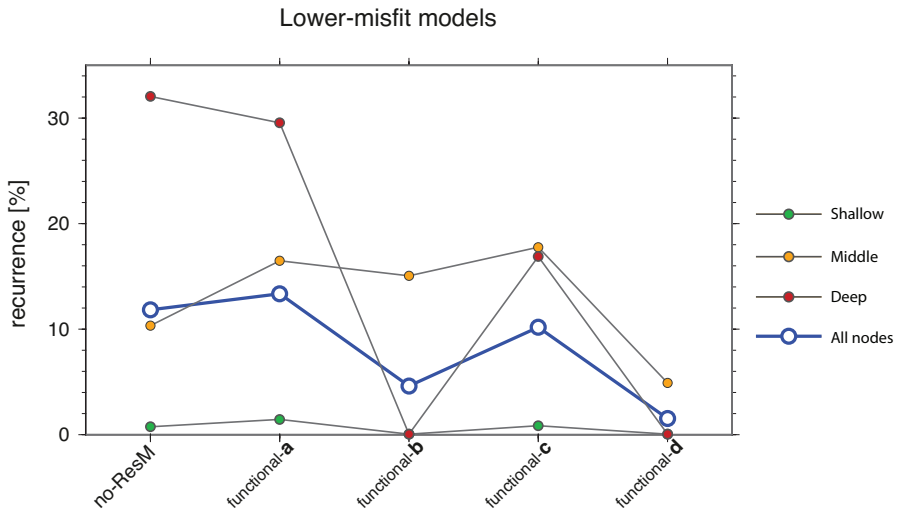


Figure 3.12: *Misfit analysis: percentage of models with a reduced misfit after a perturbation for five perturbation schemes (classical M-H no-ResM and 4 ResM functionals). Shallow, middle and deep sets of nodes have been analyzed separately.*

of better models proposed perturbing shallow nodes with the functional-a turned out to be almost twice as high than with functional-c or with the no-ResM functional. The remaining two functionals, b and d generally show poor performances.

Likelihood-difference distributions The second property examined in the non-McMC test is the probability distribution of the likelihood differences (L_D) between test (current) and proposed (trial) model $L_D = L_C - L_T$. It is known that in McMC processes the probability of accepting a step between "neighboring" models is higher than that of moving between completely different models. The property of two models to be neighbors, can be considered both from a *data* and from a *parameter*-perspective: two models could show minimum differences in the values taken by the respective parameters, but still produce sensibly different data under some measure. Vice-versa different models could produce the same data. Considering the similarity between models from the *data*-perspective, we can state that trial models that are more likely to be accepted in a McMC are characterized by a Likelihood-difference value around 0. In our non-McMC test we cannot talk about *accepted* and *rejected* models, nonetheless L_D-distributions can provide an helpful estimation of the performances of perturbation schemes based on different functionals.
We compared the L_D-distributions of the models proposed by the non-ResM functional with those proposed by the other functionals. A better-performing algorithm is expected to produce a rightwards-shifted L_D-distribution, with a higher recur-

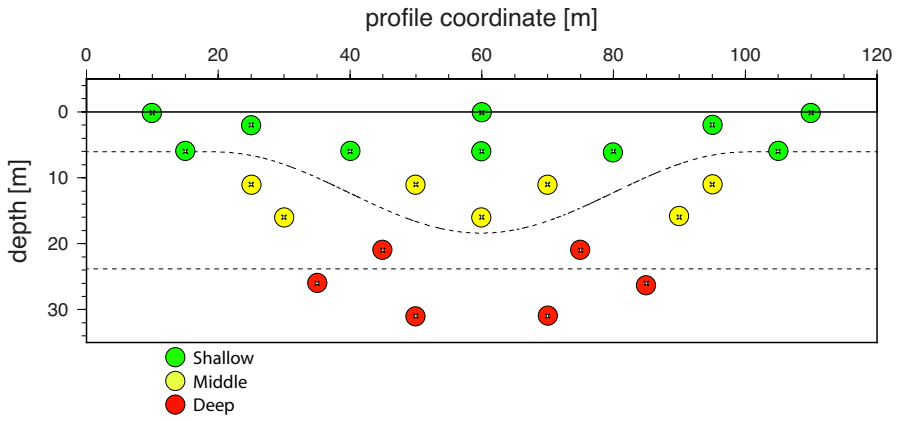


Figure 3.13: *Depth-based subdivision of the nodes in the model in three groups: shallow (green) , middle (yellow), and deep (red). The dashed lines are contour lines (as in Fig.3.3.a) reported as a reference for the synthetic structure.*

rence for values around zero. This positive behavior is found in functionals a and c (Fig.3.14) where the maximum of the distribution is a sharp peak with a higher recurrence value. The remaining distributions exhibit worse properties.

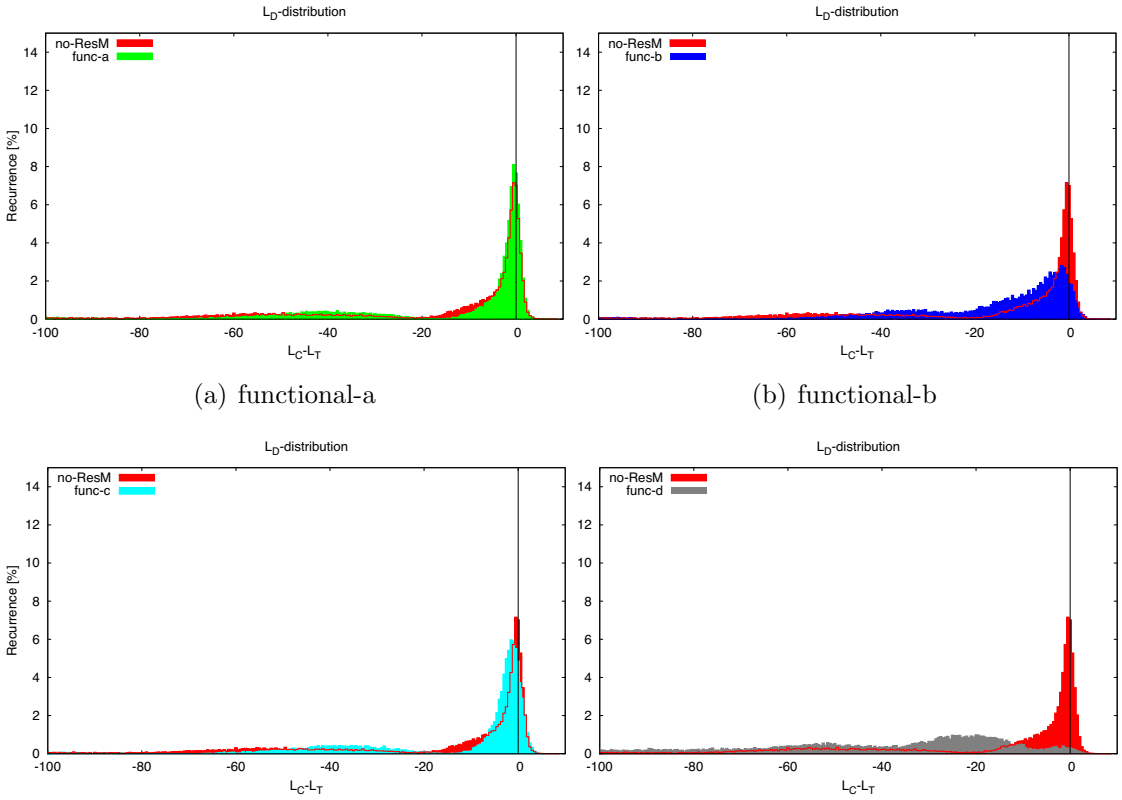
Scaling the *step size* Excluding stochastic influences was the main reason for us to perform this non-McMC test where the perturbation size can be controlled, thus allowing for a comparison between perturbation schemes, no specific limits were however set to control the resolution-matrix-based compensations. This could result in some of the functionals to produce overall larger compensations thus biasing the chance to compare "equal-sized" steps between models. To ensure that, the step (perturbation + compensations) size is defined through the $\mathbf{L}^2 - norm$ as:

$$||\Delta v|| = \sqrt{\sum_{j=1}^{n} \Delta v_j^2} \tag{3.29}$$

and a condition is imposed so that the step sizes for all the perturbation schemes, scaled by a factor γ, must be equal to the non-ResM step size Δv_i :

$$\gamma||\Delta v|| \stackrel{!}{=} \Delta v_i \tag{3.30}$$

Imposing such condition the non-mcmc test was repeated and the same analysis was performed on the *scaled*-dataset. The statistical properties of misfit and L_D-distributions show substantially no difference with the *unscaled*-dataset, hence no



(a) functional-a (b) functional-b

(c) functional-c (d) functional-d

Figure 3.14: *L_D-distributions produced by the four functional under exam compared with the distribution obtained with the no-ResM functional (in red). A better-performing algorithm is expected to produce a rightwards-shifted L_D-distribution, with a higher recurrence for values around zero.*

influence on the performances of the tested updating schemes is to be expected due to inherent differences in the proposed step sizes.

3.3.2 Fix-ResM McMC test

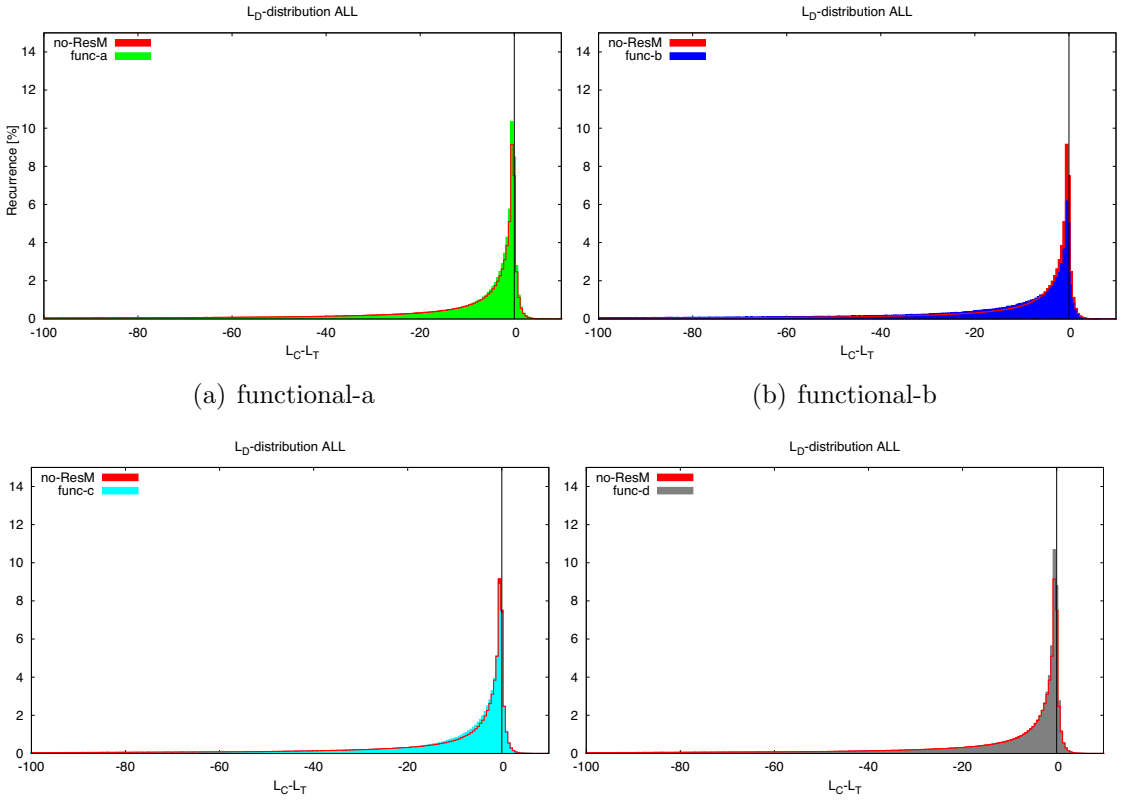
The testing phase to compare the four different functionals (3.27a/3.27d) proceeded with the analysis of the ensemble properties of Markov chains obtained employing the Fix-ResM McMC each one characterized by a different functional. For the non-McMC test we started analyzing misfit and L_D-distributions, we will proceed similarly with the only difference that in a McMC contest we can now consider the acceptance rate instead of it's misfit-based estimator we used in the previous section (Fig.3.12).

no-ResM	**func-a**	**func-b**	**func-c**	**func-d**
24.50	27.71	17.17	24.40	27.76

Table 3.3.1: *Acceptance rates* [%] *relative to Markov chains characterized by the use of the four functionals plus the "classical" non-ResM McMC.*

Acceptance rates The no-ResM McMC we are looking to improve accepted 24.50% of the proposed models, an higher acceptance rate is shown in the Markov chains based on the functionals-a and d with a value around 27.7% (table.3.3.1). The functional-b based McMC results in a drastic decrease in the acceptance rate whereas the remaining functional-c shows a value comparable with the "classical" algorithm.

Likelihood-difference distributions Following the same procedure applied for the non-McMC tests the likelihood differences were analyzed confirming the behavior observed in Fig.3.14 which highlighted the functional-a as the best candidate able to shift part of the original non-ResM distribution (in red in Fig.3.15) to obtain a sharper peak around zero. Acceptance rate values and L_D-distributions show a quantitative agreement while pointing out functionals-a and d as the most appropriate candidates.
A good qualitative and quantitative agreement is found also comparing L_D - distributions in Fig.3.15, proving the ability of the non-stochastic tests to independently evaluate and foresee some of the statistic properties of the Markov chains. An exception should be made for *functional-d* (Fig.3.14.d and 3.15.d) that displays an incongruence in the behavior of its L_D-distributions. A possible explanation could lie in the nature itself of our non-McMC test: we decided to remove stochastic influences defining a fixed perturbation magnitude (10% of the prior range) which



(a) functional-a (b) functional-b

(c) functional-c (d) functional-d

Figure 3.15: *Comparison of the L_D-distributions of Markov chains based on the Fix-ResM updating scheme. In red the reference distribution relative to the "classical" McMC.*

might have caused the *functional-d* to produce compensations that resulted quite often in "worse" proposed models (i.e. L_D-distribution less-peaked around zero). Testing the same functional in a McMC allows instead the perturbation magnitudes to be randomly selected on a continuous set, thus eliminates this behavior observed in the non-stochastic test. In other words, while our non-McMC test provides a useful tool to compare the behaviors of different perturbation schemes, one should be aware that the statistical population analyzed is way more limited than that of an actual McMC, thus could be subject to slightly different outcomes. More attention could be dedicated to this point in future studies. Anyway the results obtained with the non-McMC tests as we described them, seem to ensure a good level of reliability and help in the analysis of the different functionals proposed.

Mixing properties The third of the three criteria we outlined in section 3.1.2 states that a Markov chain is better if it allows a faster sampling of the state

space. Instead of computing an estimate of the expectation (eq.3.3) a graphical evaluation of the sampling speed of a Markov chain can be conveniently achieved with a trace plot of the values taken by parameters or functionals of the chain itself. Two model parameters were selected, corresponding to one superficial node (depth $0.1m$) and a deep node (depth $25m$), the trace plots report the P-velocity value assumed by the parameters at each McMC step for a total of 3000 iterations. The non-ResM chain (uppermost row of Fig.3.16) displays a good mixing for the bottom node: there is a good balance between the frequency of accepted moves (appropriate acceptance rate) and the change in the velocity values (appropriate step size). On the contrary the upper node has poor mixing properties with a low frequency of accepting moves (low acceptance rate) and relatively big changes. One possible interpretation of this behavior could consider it a consequence of a still sub-optimal scaling of the perturbation size, despite the depth-dependent perturbation scaling applied (see section 3.2.5). A second, non-complementary way to interpret the behavior of the shallow nodes is to observe that they are relatively well-constrained, even small perturbations have a large impact on the data, which makes perturbations less likely to ba accepted. This group of nodes is the one where we expect the most improvement. Applying our resolution matrix-based updating scheme we can compare the different effects of the four tested functionals on the mixing properties of the resulting Markov chains:

a) The upper node shows an improved mixing, the frequency of accepted moves has increased together with the step size. No substantial improvement can be visually appreciated for the deep node.

b) Strongly improved mixing can be observed for the shallow node, on the other hand there's a drastic decrease of performance regarding the deeper node where we observe a high frequency of accepting small velocity changes resulting in a inefficient sampling.

c) We observe a strongly improved mixing for the shallow node, and a slight decrease in sampling efficiency for the lower nodes.

d) The shallow node exhibits poor mixing ability at a level comparable with the non-ResM chain. The deep node instead seem to improve the sampling performances.

Summarizing what observed so far, the *functional-a* appears to improve the mixing ability for parameters located at all depths while the other functionals seem to have worse performances either for deep (*functional-b* and *c*) or for surface parameters (*functional-d*). Similarly, also trace plots for mid-depth parameters have been analyzed, verifying that improvements to the mixing properties can be observed

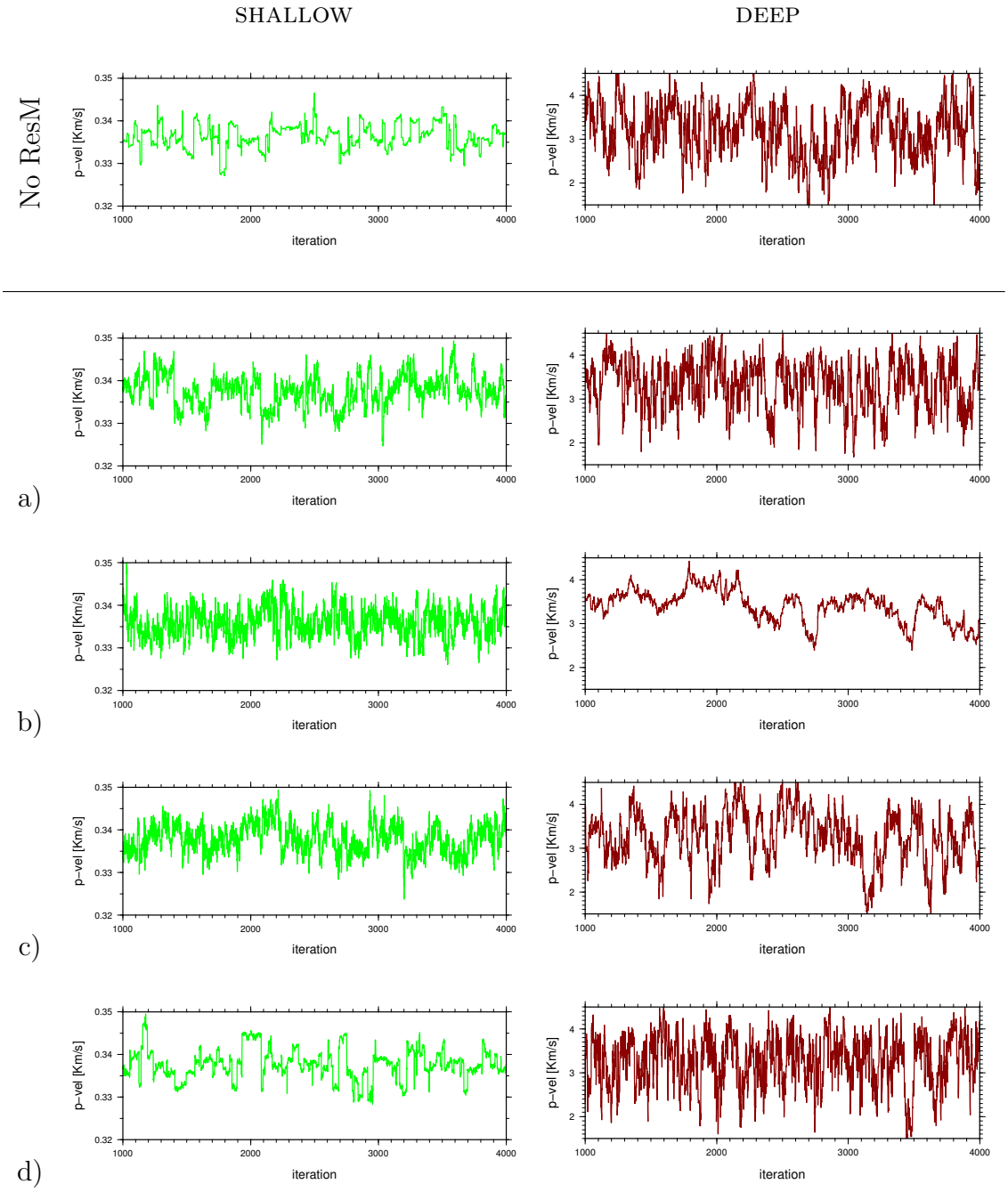


Figure 3.16: *McMC traces of two parameters corresponding to nodes at depths of respectively 0.1 and 25 m. The P-velocity is plotted for every McMC iteration in the range* $[1000, 4000]$*. The amount and frequency of velocity changes provide a qualitative estimation of the mixing properties of each algorithm. The difference is especially strong for the uppermost parameters, here it's clear that the use of our FIX-ResM updating scheme (green traces on the right) results in a more frequent update of the velocity value.*

from all the functionals at a comparable level thus we won't report any trace plot for such nodes.
To support quantitatively what has been qualitatively deduced from the McMC trace plots we can estimate values of eq.(3.3) by means of the step size given in eq.(3.29) as:

$$E_{\|\Delta v\|} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{n}\Delta v_j^2\right)} \tag{3.31}$$

where N is the number of the steps attempted during the Markov process and n the number of model parameters. Such estimator provides a quantification of the mixing ability of a Markov chain in a form that resembles a quadratic mean of the step sizes of a chain. Evaluations of eq.(3.31) are reported in table 3.3.2:

	SHALLOW	MIDDLE	DEEP
no-ResM	0.007280	0.052309	0.192393
functional-a	0.011934	0.072848	0.197953
functional-b	0.009932	0.061792	0.035045
functional-c	0.011876	0.081926	0.116228
functional-d	0.008493	0.065230	0.236194

Table 3.3.2: *Evaluations of eq. 3.31 computed for each functional.*

Here the functional-a proves once more to lead to a Markov chain with improved mixing properties at all depths, especially for shallow, highly constrained model parameters. It's evident that also the functional-d improves the mixing at all depths especially for the deepest nodes where it performs better than the other candidates.

Choice of *functional-a* Considering the outcome of the tests conduced we selected *functional-a* as the best of the examined candidates. If this choice is clear considering *functionals-b* and *c*, the last option *d* appears to perform on a similar level, then a justification is needed. While "global" indicators as acceptance rate and L_D-distributions gave substantially similar results for both *a* and *d*, the graphical and quantitative analysis of mixing show how the two functionals perform differently. *Functional-d* leads to an improved mixing for all the nodes but the gain is more focused on deep nodes, thus on less constrained model parameters. In the framework of seismic tomography it appears however more desirable to obtain a better mixing of highly constrained parameters. From a sampling point of view improving the mixing of strongly constrained nodes translates in an increased

acceptance of moves involving perturbations to parameters that have a stronger influence on the likelihood.
Furthermore as expected "better" sampling leads to a reduced variance of the posterior estimates (eq.3.2), in other words the PDFs obtained from better chains are supposed to have smaller variance after the same number of iterations. Figure 3.17.*a* shows that the difference of the variances between the *functional-a*-chain and the no-ResM-chain is basically only negative, contrary to the other maps that show also increases in the variance. This behavior is quantitatively limited for the simple test model utilized but is expected to become more evident for other more complex models.

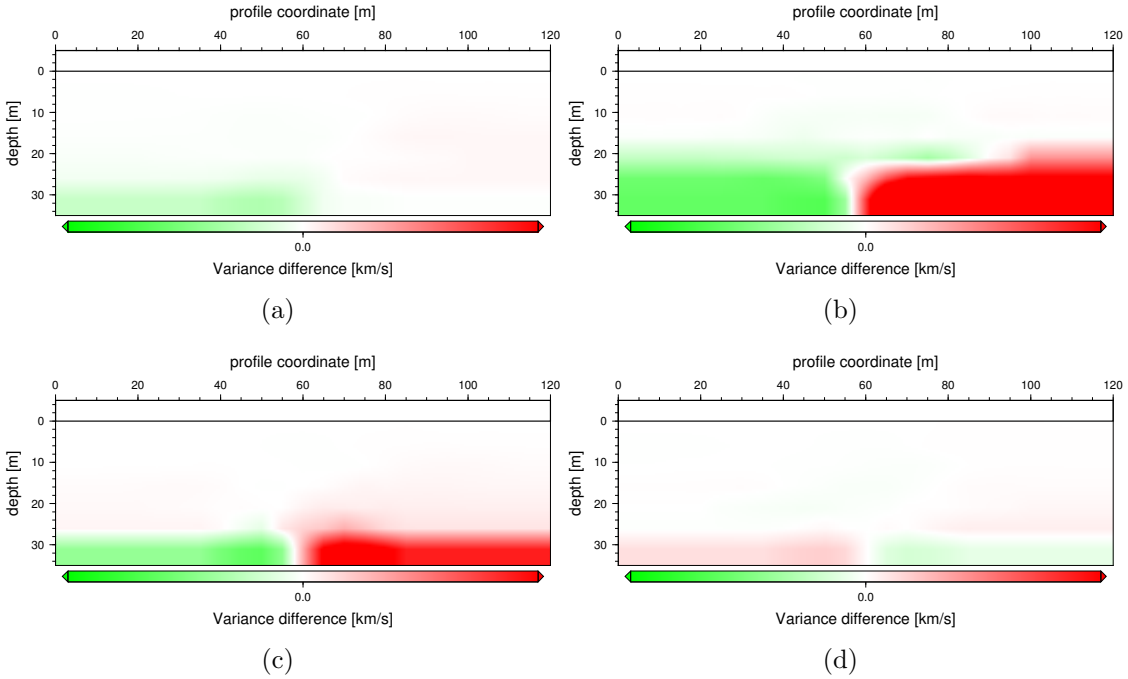


Figure 3.17: *Variance difference maps between the four functionals and the NoResM McMC. Functional-a is the only that displays only a variance reduction.*

3.3.3 Bayesian seismic tomography with the Fix-ResM McMC algorithm

Functional-a, expressed in eq.3.27a, was selected as the best of the proposed candidates, this choice was based on the criteria introduced in section 3.1.2, and investigated with non-stochastic and McMC tests.
We run the fix-ResM McMC algorithm fitted-out with the chosen functional on a desktop workstation equipped with an eight-core *Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz* for 20 days. A total of $\approx 3.63M$ iterations produced a set of $1.0M$ models, every 10^{th} accepted model has been stored as state of the Markov chain. Thinning the chain allows to ease the computational load in the analysis phase of the statistical ensemble avoiding to deal with an unnecessary huge number of highly correlated models. No burn in was needed since, with the choice of the DLS-solution as initial model state, the Markov chain was initialized near the center of the equilibrium distribution.

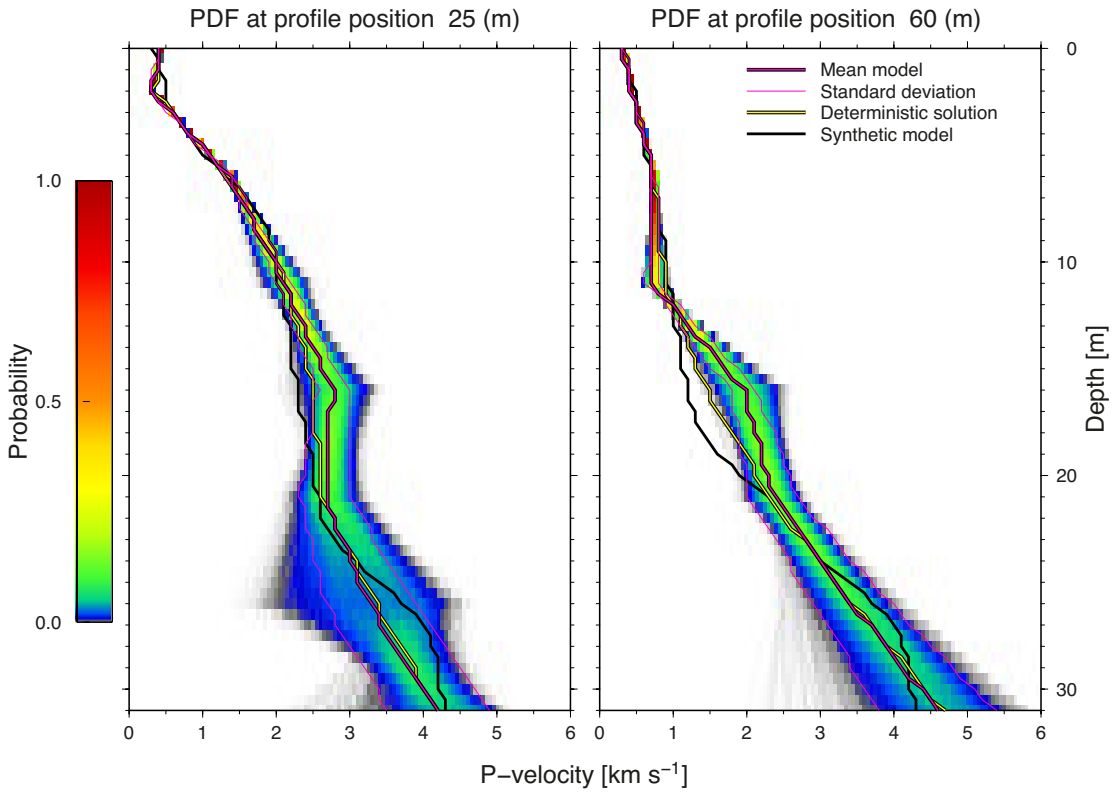


Figure 3.18: *Vertical cross sections of the posterior distribution at profile position 25 and 60 m. displayed together with the mean model, DLS deterministic solution and synthetic "real" values. The thin pink lines mark the confidence interval given by ± one standard deviation.*

The solution of a bayesian inversion is the posterior distribution, a probability density function (PDF) that holds all the statistical information regarding the sampled ensemble of models. The estimated values of velocity are given for every parameter together with the relative probability.

3.3.3.1 Ensemble properties

Mean model The mean model was extracted as a spatial average of the posterior distribution: the mean of the velocity value assumed by each model parameter is computed from the posterior distribution through equation 1.35. Normally, while averaging the sampled ensemble, one should account only for the post-burn in models but, once again, the particular initial model-state we chose demands no burn-in phase.

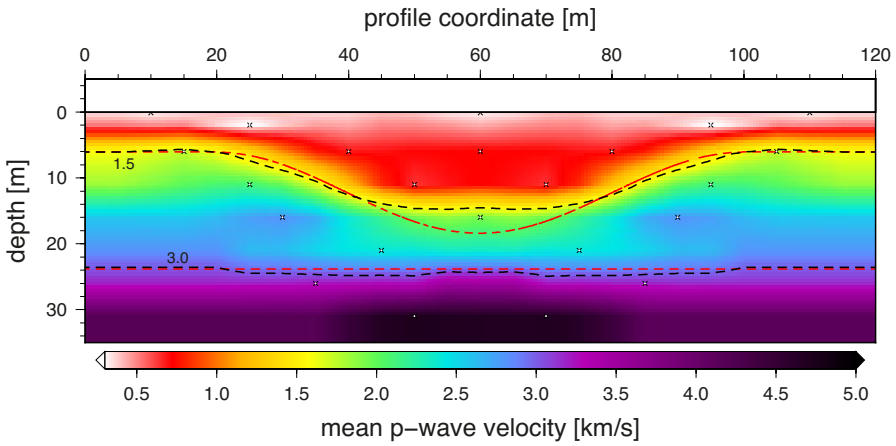


Figure 3.19: *Mean model: the dashed black line represents the isolines corresponding to p-wave velocities of 1.5 and 3.0 Km/s of the extracted mean-mode solution, the red dashed reports for comparison the same isolines for the test synthetic model.*

Fig.3.19 shows that the synclinal upper structure of the test model is well reproduced with the exception of the central part correspondent to the node located 16 m. deep at 60 m. (profile coordinate). The lower layer that extends in depth from 22 m. is correctly reproduced.

Uncertainty visualization Similarly to what done for the mean model, an uncertainty map was extracted from the posterior distribution by means of eq.2.20, obtaining a spatial map for the standard deviation of the sampled ensemble. Comparing this standard deviation with the resolution diagonal elements map of the deterministic solution (Fig.3.20) we can observe a good qualitative agreement despite the completely different origin of the two uncertainty-estimators: the stan-

dard error map is in fact a posterior estimate obtained from the sampled models while the RDE map is computed on the last iteration solution of the DLS inversion. A fundamental difference lies however between these two methods of uncertainty visualiztion: the bayesian approach allows a quantitative estimation of the uncertainty, i.e. a measure of confidence. On the other hand the RDE map only provides qualitative values, a relative measure of where the resolution is higher, depending on the damping values utilized in the deterministic inversion process. Figure 3.20.a allows a quantification of the uncertainty that characterizes the mean model: the shallow portion with velocities ≤ 1.5 Km/s is associated with a standard deviation up to 0.1 Km/s, while the bottom part with higher velocities shows uncertainties ≥ 0.5 Km/s, value that corresponds to the isoline of 3.0 Km/s in Fig.3.19.

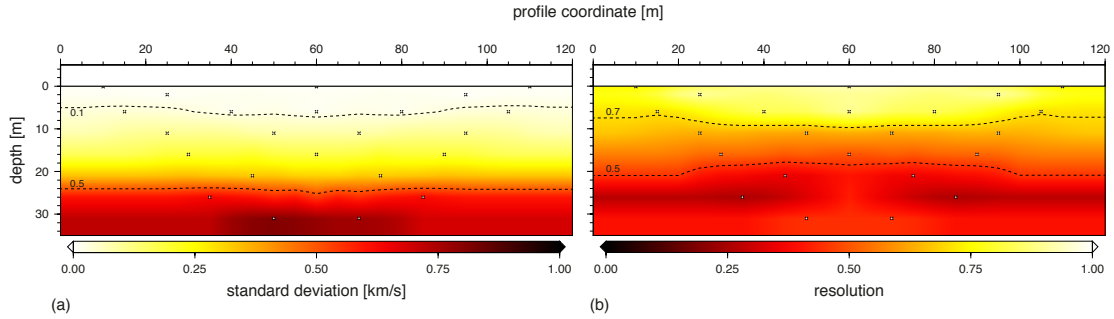


Figure 3.20: *Standard deviation map obtained from the posterior probability distribution (a) and map of the resolution diagonal elements obtained from the last iteration of the DLS solution (b). For the resolution map, the contour lines are defined basing on the node subdivision of Pag.77.*

Useful practice to highlight parameters affected by relatively higher uncertainties is a relative error map defined by means of the standard deviation and the mean model maps. This map reported in Fig.3.21 makes possible to notice that the central area of the model (the portion of the synclinal structure deeper than 6 m.) is probably affected by errors relatively high for that depth. Comparing the mean model (Fig.3.19) with the synthetic model (Fig.3.2) one can verify that the estimated mean velocity value of the two nodes located at 11 m depth and profile positions 50 and 70 m is actually lower than the synthetic-real value, while the underlying node has an estimated mean velocity higher than the synthetic value.

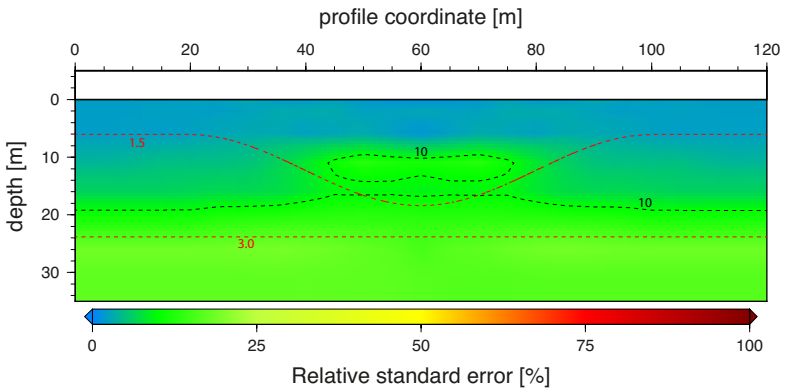


Figure 3.21: *Relative error map: the black dashed line marks the contours of 10% relative standard error, the red contours report for comparison the main structural features of the synthetic model.*

3.4 Discussion and Conclusions

Besides the application of our resolution-matrix-based multivariate updating scheme to a synthetic dataset, as reported in section 3.3.3, we employed a classical M-H McMC for the inversion of the same dataset. Both Markov chains were left running until each one accepted a total of 1 Million of models. A recap of some parameters for the two inversions is reported in Table 3.4.1.

	no-ResM	**fix-ResM**
Accepted models	1 Mil.	1 Mil.
Number of Iterations	4080293	3627027
Runtime	554.5 hrs.	497.2 hrs.
Acceptance rate	24.508 %	27.571 %

Table 3.4.1: *Performance comparison between a classical M-H McMC and our Fix-ResM algorithm.*

We observed that in the Fix-ResM Markov Chain the acceptance ratio increased by about 3% compared to the inversion that makes use of a classical updating scheme. On the usual time scale of bayesian inversions this can translate in a reduction of the computation time that in the case we illustrated corresponds to $\approx 10\%$.
The McMC-trace analysis shows that our multivariate updating scheme results in wider and more frequent movement of the parameters values. In addiction the increase in the acceptance ratio without diminishing the perturbation size leads us to the conclusion that, using such a ResM-based perturbation scheme, a McMC will have better mixing properties, thus will sample the model space with higher efficiency.

Chapter 4

Discussion and conclusions

4.1 Achievements

The research we conducted in this project aimed to face some of the major issues of geophysical inversion processes such as the quantification of the underdetermination that affects inverse solutions, the choice of a parametrization, in terms of both node density and geometry, the determination of some inversion parameters as damping and smoothing. These issues have been tackled with a Bayesian approach, through the implementation of Markov chain Monte Carlo inversion algorithms. The intrinsic intensive computational load that characterizes this family of sampling methods demands for optimized procedures. We focused our attention on two different optimization strategies that have been applied respectively on transdimensional and non-transdimensional inversion schemes: parallelization and optimization of transition kernels.

Trandimensional McMC We have proved that despite the intrinsic serial nature of Markov chains, it is possible to exploit the benefits offered by parallel computing with a trivial implementation that requires minimal coding effort. The practice of running independent parallel Monte Carlo sampling processes showed its effectiveness together with the inversion algorithm we developed also thanks to the extremely limited portion, as small as 0.5%, of models discarded within the burn-in phase. This was achieved initializing the Markov processes with a deterministic DLS solution, however even chains initialized with fairly simple velocity models, as linear gradients, produced limited burn-in phases. This results in an almost linear dependence between speedup and CPU/cores involved in the parallelization, thus in the opportunity to make use of a massive parallelization to gain a more efficient sampling.
The use of a staggered-grid approach in addition to independent parallel transdi-

mensional Markov chains allowed to recover more structural details and smoother reference solution models than a deterministic DLS inversion. The mean model map of the Salzach valley extracted from the sampled ensemble exhibits a high level of compatibility with the refraction-reflection (RRTT) and with the full-waveform solutions of Bleibinhaus and Hilberg (2012) despite the use of information carried by first-breaks traveltimes only. The higher computation cost of this Bayesian approach seems therefore to be repaid not only by the possibility to provide a quantitative estimation of the modeling errors, but also from solution models able to recover smaller structural details.

Resolution-Matrix-based updating scheme We developed a multivariate updating scheme that utilizes the Model Resolution Matrix to propose trial models with a higher probability of acceptance. The Bayesian inversion algorithm we developed is based on the Metropolis-Hastings scheme, a trial model is here proposed applying a normal-distributed *main* perturbation to one parameter of the current model-state, followed by compensations applied to all the remaining model parameters. Such compensations, computed by means of a functional of the Resolution Matrix are aimed to reduce the global biasing effect of the *main* perturbation. The resulting updating scheme proved to lead to better performing Markov chains in terms of improved mixing properties, reduced variance of the sampled ensemble, increased acceptance rate.
While multivariate proposal schemes find their main limitation in the difficulty to propose trial models yielding a high acceptance probability, we proved that the Model Resolution Matrix can be successfully utilized to overcome such issue.

4.2 Future directions

In our quest for optimal McMC inversion algorithms we would like to focus on some of the aspects of the algorithms we developed, that could have room for further improvement.

4.2.1 Voronoi parametrization

The node-based parametrization and the interpolation scheme characterizing the `simulr16` code, allow for the reconstruction of complex structures with a relatively limited number of model parameters. The consequent restrained cardinality of the model space is undoubtedly a benefit for the Bayesian inversion algorithms presented in this work since the runtime needed to sample a model space increases with the number of its elements.

However a downside of this model parametrization is the fact that transdimensional McMC inversions won't fully benefit from the Occam's razor principle that naturally characterizes this family of McMC methods. A model parametrization defined through a precisely limited set of nodes, with defined positions could somehow appear as a contradiction with the transdimensional approach that would on the contrary prefer to have total freedom in the choices of position and number of inverse parameters during the sampling process. A well diffused parametrization strategy that satisfies these characteristics is based on the Voronoi tessellation, some of the latest and most notable applications on seismic tomographic inversion can be found in Bodin and Sambridge (2009); Bodin et al. (2012). Through Voronoi polygons, or Delaunay triangles one could parametrize models allowing for non-discretized node positions, thus independent from underlying pre-defined gridded locations. An interesting direction of study could be the comparison of the performances of McMC tomographic inversions, within the `simul` framework, associated to different parametrization strategies such as the mentioned Voronoi and Delaunay, considering as well different interpolation algorithms.

4.2.2 Reflection-refraction seismics

A Bayesian approach could be extended to the analysis of reflection data as well. `Simulr16` provides already provides established tools for the deterministic inversion of such refraction-reflection data sets. Reflecting interfaces are treated in `simulr16` with a method of Bleibinhaus (2003), based on a nearest neighbor interpolation. Reflectors are parameterized as splines on separate grids, and the corresponding velocity discontinuities arise from interrupting the velocity interpolation. That allows for defining smooth, curved reflector surfaces with very few parameters. The modeling of reflectors in this method is highly flexible: they can be floating, i.e. without connection to the velocity field, or they can be discontinuities. Also, because the parameterization is not layer-based, reflectors must not span the entire model, nor have they to be sub-horizontal. All the parameters involved in the definition of reflectors could be treated as unknowns within a McMC inversion process: depths, coordinates as well as the number of nodes that define a reflector could be therefore described in terms of posterior distributions expressing the probability for an interface to have certain characteristics (i.e. shape, depth...). This method would require the formal definition of prior distributions and likelihood functions providing an appropriate probabilistic description of the quantities connected with reflectors.

4.2.3 Combined functionals

For our fix-ResM updating scheme we proposed and compared four different functionals of the Model Resolution Matrix, analyzing their performances both in a McMC environment and non-stochastic tests. From the results we obtained it appears that the performances of the tested functionals are partially dependent on the respective RDE value of each inverse parameter: while the mixing properties of highly constrained nodes (i.e. with RDE values approaching the unity) shows the highest increment with one functional, another leads to an optimal mixing for less constrained parameters. We reckon that more functionals should be proposed and tested seeking for an overall "best performer". In the absence of a clear optimal choice a promising strategy appears to be the combination of the best functionals on RDE basis. One could make use of multiple functionals at the same time: let's assume that three functionals α, β, γ have been identified as the most appropriate respectively for highly, medium, and poorly-constrained inversion parameters. In our fix-ResM algorithm, a perturbation of a highly-constrained parameter would be followed by compensations computed using the α-functional. Similarly, perturbations of medium-constrained nodes would be associated to β-functional compensations and γ for the highly-constrained nodes. This approach brings however the question "how to define highly/middle/low constrained nodes?". Thresholds would need to be defined for the RDE values characterizing each of the groups, with the consequent issues connected to this arbitrary choice. An alternative strategy could be a continuous approach that makes use of all the three functionals at once, assigning them RDE-dependent weighting factors, allowing therefore for "smoothed boundaries" between parameter groups. Along these lines a trial model would be proposed with a slightly different version of (3.18) as:

$$\mathbf{m}' = \mathbf{m} + u\sigma_i \mathbf{e}_i - u\sigma_i \sum_{j \neq i}^{n} \left[w_\alpha g_\alpha(R_{ij}) + w_\beta g_\beta(R_{ij}) + w_\gamma g_\gamma(R_{ij}) \right] \mathbf{e}_j \tag{4.1}$$

where w_α, w_β, w_γ are weighting factors such that $w_\alpha + w_\beta + w_\gamma = 1$, proportional to the RDE value R_{ii} of the i-node being perturbed. In this way one could obtain an updating scheme where the resolution-matrix-based compensations are always computed with an *optimal* functional.

4.2.4 Transdimensional McMC and Resolution Matrix

So far we treated transdimensional and fixed-dimension McMC algorithms as two distinctly separated groups. The two updating schemes we implemented are at the actual stage non compatible with each other, it appears therefore of high interest to research the feasibility of utilizing the Resolution Matrix to produce compensations in a multivariate updating scheme within a transdimensional framework.

4.2.5 A unified approach

For the McMC-optimization perspective we adopted so far it is of fundamental importance to join all the possible strategies that could contribute to the generation of optimal Markov chains, to obtain a fast and reliable working tool for the Bayesian inversion of seismic tomographic data. Multiple instances of the sampler should run in parallel on independent cores/CPUs, the forward solver should be parallelized over the sources, and a staggered grid approach should be used in combination with the `simulr16` built-in node-parametrization. In case a different parametrization would be used, based for instance on Voronoi or Delaunay tessellation, the staggered approach could be avoided. One should initialize the Markov processes with a deterministic solution model in order to sensibly shorten or even eliminate the need to throw away models in the burn-in.
The most crucial problem in the application of McMC methods lies in the optimization of the algorithms, in order to obtain a maximum reduction of the computation-time, given the limits typically imposed by the available time and hardware resources. It is therefore desirable that all the established (and compatible) methods, which could contribute to increase efficiency and quality of Bayesian inversion algorithms, will be employed together in a framework offered by an highly parallelized code whose forward and inverse routines are optimized to run on clusters of variable dimensions.

Bibliography

Ajo-Franklin, J., Urban, J., and Harris, J. (2006). Using resolution-constrained adaptive meshes for traveltime tomography. *Journal of Seismic Exploration*, 14:371–392.

Bleibinhaus, F. (2003). *Entwicklung einer simultanen refraktions- und reflexionsseismischen 3D-Laufzeittomographie mit Anwendung auf tiefenseismische TRANSALP-Weitwinkeldaten aus den Ostal- pen.* PhD thesis, Ludwig-Maximilians-Universität München.

Bleibinhaus, F. and Gebrande, H. (2006). Crustal structure of the eastern alps along the transalp profile from wide-angle seismic tomography.

Bleibinhaus, F. and Hilberg, S. (2012). Shape and structure of the salzach valley, austria, from seismic traveltime tomography and full waveform inversion. *Geophysical Journal International*, 189(3):1701–1716.

Bleibinhaus, F., Hilberg, S., and Stiller, M. (2010). First results from a seismic survey in the upper salzach valley, austria. *Austrian Journal of Earth Sciences*, 103(2):28–32.

Bodin, T. and Sambridge, M. (2009). Seismic tomography with the reversible jump algorithm. *Geophysical Journal International*, 178(3):1411–1436.

Bodin, T., Sambridge, M., Rawlinson, N., and Arroucau, P. (2012). Transdimensional tomography with unknown data noise. *Geophysical Journal International*, 189(3):1536–1556.

Böhm, G., Galuppo, P., and Vesnaver, A. (2000). 3d adaptive tomography using delaunay triangles and voronoi polygons. *Geophysical Prospecting*, 48(4):723–744.

Brockwell, A. E. (2006). Parallel markov chain monte carlo simulation by prefetching. *Journal of Computational and Graphical Statistics*, 15(1):246–261.

Brooks, S., Gelman, A., Jones, G. L., Meng, X.-L., and Rosenthal, J. S. (2011). *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC.

Campillo, F., Rakotozafy, R., and Rossi, V. (2009). Parallel and interacting markov chain monte carlo algorithm. *Mathematics and Computers in Simulation*, 79(12):3424–3433.

Dębski, W. (2010). Seismic tomography by monte carlo sampling. *Pure and Applied Geophysics*, 167(1-2):131–152.

Del Moral, P. (1996). Non linear filtering: Interacting particle solution. *Markov Processes and Related Fields*, 2(4):555–580.

Duijndam, A. J. W. (1988a). Bayesian estimation in seismic inversion. part i: principles. *Geophysical Prospecting*, 36(8):878–898.

Duijndam, A. J. W. (1988b). Bayesian estimation in seismic inversion. part ii: uncertainty analysis. *Geophysical Prospecting*, 36(8):899–918.

Eberhart-Phillips, D. (1986). Three-dimensional velocity structure in northern california coast ranges from inversion of local earthquake arrival times. *Bulletin of the Seismological Society of America*, 76(4):1025–1052.

Gallagher, K., Charvin, K., Nielsen, S., Sambridge, M., and Stephenson, J. (2009). Markov chain monte carlo (mcmc) sampling methods to determine optimal models, model resolution and model choice for earth science problems. *Marine and Petroleum Geology*, 26(4):525 – 535. Thematic Set on Basin Modeling Perspectives.

Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models (comment on article by browne and draper). *Bayesian Analysis*, 1(3):515–534.

Gelman, A., Roberts, G. O., and Gilks, W. R. (1996). Efficient metropolis jumping rules. *Bayesian Statistics*, 5:599–607.

Gelman, A. and Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statist. Sci.*, 7(4):457–472.

Geyer, C. J. (1991). Markov chain monte carlo maximum likelihood. *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, pages 156–163.

Geyer, C. J. (1992). Practical markov chain monte carlo. *Statistical Science*, 7(4):pp. 473–483.

Geyer, C. J. and Møller, J. (1994). Simulation procedures and likelihood inference for spatial point processes. *Scandinavian Journal of Statistics*, 21(4):359–373.

Gilks, W. R. (2005). *Markov Chain Monte Carlo*. John Wiley and Sons.

Green, P. J. (1995). Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732.

Green, P. J., Hjort, N. L., and Richardson, S. (2003). *Highly Structured Stochastic Systems*, volume 6 of *27*. Oxford Statistical Science Series.

Green, P. J. and Mira, A. (2001). Delayed rejection in reversible jump metropolis–hastings. *Biometrika*, 88(4):1035–1053.

Gubbins, D. (2004). *Time Series Analysis and Inverse Theory for Geophysicists*. Cambridge.

Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109.

Hole, J. A. (1992). Nonlinear high-resolution three-dimensional seismic travel time tomography. *Journal of Geophysical Research: Solid Earth*, 97(B5):6553–6562.

Hole, J. A. and Zelt, B. C. (1995). 3-d finite-difference reflection travel times. *Geophysical Journal International*, 121(2):427–434.

Keilis-Borok, V. I. and Yanovskaja, T. B. (1967). Inverse problems of seismology (structural review). *Geophysical Journal of the Royal Astronomical Society*, 13(1-3):223–234.

Koutsourelakis, P.-S. (2009). Accurate uncertainty quantification using inaccurate computational models. *SIAM Journal on Scientific Computing*, 31(5):3274–3300.

Laskey, K. B. (2003). Population markov chain monte carlo. *Machine Learning*, 50:175.196.

Levin, D. A., Peres, Y., and Wilmer, E. L. (2006). *Markov Chains and Mixing Times*. American Mathematical Society.

Marinari, E. and Parisi, G. (1992). Simulated tempering: A new monte carlo scheme. *EPL (Europhysics Letters)*, 19(6):451.

Martin A. Tanner, W. H. W. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 82(398):528–540.

Menke, W. (2012). *Geophysical data analysis : discrete inverse theory*. Elsevier, 3rd edition.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

Mosegaard, K. (1998). Resolution analysis of general inverse problems through inverse monte carlo sampling. *Inverse Problems*, 14(3):405.

Mosegaard, K. and Tarantola, A. (1995). Monte carlo sampling of solutions to inverse problems. *Journal of Geophysical Research: Solid Earth*, 100(B7):12431–12447.

Nolet, G. (2008). *A Breviary of Seismic tomography*. Cambridge University Press.

Pearse, S., Hobbs, R., and Bosch, M. (2009). Using a local monte carlo strategy to assess 1-d velocity models from wide-angle seismic travel-time data and application to the rockall trough. *Tectonophysics*, 472:284–289.

Press, F. (1968). Earth models obtained by monte carlo inversion. *Journal of Geophysical Research*, 73(16):5223–5234.

Rietbrock, A. (1996). *Entwicklung eines Programmsystems zur konsistenten Auswertung grosser seismologischer Datensatze mit Anwendung auf die Untersuchung der Absorptions-struktur der Loma-Prieta-Region Kalifornien.* PhD thesis, Ludwig-Maximilians-Universität München.

Robert, C. and Casella, G. (2004). *Monte Carlo statistical methods*. Springer.

Roberts, G. O., Gelman, A., and Gilks, W. R. (1997). Weak convergence and optimal scaling of random walk metropolis algorithms. *Ann. Appl. Probab.*, 7(1):110–120.

Roberts, G. O. and Rosenthal, J. S. (2007). Coupling and ergodicity of adaptive markov chain monte carlo algorithms. *J. Appl. Probab.*, 44(2):458–475.

Rosenthal, J. S. (2000). Parallel computing and monte carlo algorithms. *Far East Journal of Theoretical Statistics*, 4(2):207–236.

Sambridge, M. (2014). A parallel tempering algorithm for probabilistic sampling and multimodal optimization. *Geophysical Journal International*, 196(1):357–374.

Sambridge, M., Braun, J., and McQueen, H. (1995). Geophysical parametrization and interpolation of irregular data using natural neighbours. *Geophysical Journal International*, 122(3):837–857.

Sambridge, M., Gallagher, K., Jackson, A., and Rickwood, P. (2006). Trans-dimensional inverse problems, model comparison and the evidence. *Geophysical Journal International*, 167(2):528–542.

Sambridge, M. and Mosegaard, K. (2002). Monte carlo methods in geophysical inverse problems. *Reviews of Geophysics*, 40(3):3–1–3–29. 1009.

Shapiro, N. M. and Ritzwoller, M. H. (2002). Monte-carlo inversion for a global shear-velocity model of the crust and upper mantle. *Geophysical Journal International*, 151(1):88–105.

Socco, L. V. and Boiero, D. (2008). Improved monte carlo inversion of surface wave data. *Geophysical Prospecting*, 56(3):357–371.

Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM.

Tarantola, A. (2006). Popper, bayes and the inverse problem. *Nature Physics*, 2(8):492–494.

Tarantola, A. and Valette, B. (1982). Inverse problems = quest for information. *Journal of Geophysics*, 50(3):150–170.

Thurber, C. and Eberhart-Phillips, D. (1999). Local earthquake tomography with flexible gridding. *Computers and Geosciences*, 25(7):809 – 818.

Thurber, C. H. (1983). Earthquake locations and three-dimensional crustal structure in the coyote lake area, central california. *Journal of Geophysical Research: Solid Earth*, 88(B10):8226–8236.

Thurber C. H., Aki, K. (1987). Three-dimensional seismic imaging. *Annual Review of Earth and Planetary Sciences*, 15:115–139.

Trinks, I., Singh, S. C., Chapman, C. H., Barton, P. J., Bosch, M., and Cherrett, A. (2005). Adaptive traveltime tomography of densely sampled seismic data. *Geophysical Journal International*, 160(3):925–938.

Tweedie, R. L. and Meyn, S. P. (1993). *Markov Chains and Stochastic Stability*. Springer-Verlag.

Um, J. and Thurber, C. (1987). A fast algorithm for two-point seismic ray tracing. *Bulletin of the Seismological Society of America*, 77(3):972–986.

VanDerwerken, D. N. and Schmidler, S. C. (2013). Parallel markov chain monte carlo. arXiv:1312.7479.

Vidale, J. E. (1990). Finite-difference calculation of traveltimes in three dimensions. *Geophysics*, 55(5):521–526.

Zelt, C. A. and Barton, P. J. (1998). Three-dimensional seismic refraction tomography: A comparison of two methods applied to data from the faeroe basin. *Journal of Geophysical Research: Solid Earth*, 103(B4):7187–7210.

Zucchini, W. and MacDonald, I. L. (2009). *Hidden Markov Models for Time Series: An Introduction Using R*. Chapman and Hall/CRC.

Acknowledgements

First of all I would like to thank my supervisor, Prof. Dr. Florian Bleibinhaus, that entrusted me with this research project. He always supported me with constructive criticism and guided me throughout the whole process, from its beginnings in the University of Salzburg, to this thesis work.

I would like to thank all the staff of the Institute of Geosciences of the University of Jena and in particular the Director Prof. Dr. Georg Büchel.

I owe a great debt of gratitude to my colleague Dr. Marco Paschke for all the helpful discussions we had and for his active advice on scripting and coding issue. I'd like also to thank M.Sc. Hamed Gharibdoost and M.Sc. Mauro Alivernini for their contribution to a positive and friendly working climate.

Last but not least I want to thank my family: my parents Silvio and Laura for their love and comprehension and my girlfriend Maria for sharing these years with me.

This research was conducted within the framework of the project P23748 "*Probabilistic uncertainty estimation for 2D/3D refraction seismic traveltime tomography*", founded by the Austrian Science Fund (FWF).

Selbständigkeitserklärung

Ich erkläre, dass ich die vorliegende Arbeit selbständig und unter Verwendung der angegebenen Hilfsmittel, persönlichen Mitteilungen und Quellen angefertigt habe.

Ort, Datum

Unterschrift des Verfassers

Curriculum Vitae

Francesco Fontanini

Birth date:	08.11.1983
Place of birth:	Verona (Italy)
Nationality:	Italian

10.2016 – 06.2012	**PhD in Geophysics** Friedrich-Schiller University of Jena Topic: Probabilistic uncertainty estimation for 2D/3D refraction seismic traveltime tomography
10.2011 – 05.2012	**Postgraduate student** University of Salzburg (Austria) Topic: Probabilistic uncertainty estimation for 2D/3D refraction seismic traveltime tomography
09.2008 – 03.2011	**Master Degree in Physics** University of Trieste (Italy) Thesis: Seismic tomography for high-resolution archaeological surveys.
09.2002 – 07.2008	**Bachelor Degree in Physics** University of Padova (Italy) Thesis: Study of electromagnetic fluctuations in low-temperature magnetized plasmas.
09.1997 – 06.2002	**High School Certificate** (Maturitá scientifica) Institute "Alle Stimate", Verona (Italy)

.....